DAAL 2024 @ Paratexts in Pre-Modern Writing Cultures

The First Workshop on Data-driven Approaches to Ancient Languages

(DAAL 2024)

Proceedings of the Workshop

Colin Swaelens, Maxime Deforche, Ilse De Vos,
Els Lefever (eds.)

June 27, 2024
Ghent, Belgium

Order copies of this proceedings from:

Language & Translation Technology Team (LT3)
Groot-Brittanniëlaan 45
9000 Ghent
Belgium
Tel: +32-9-33-11-941
https://lt3.ugent.be

ISBN 9789078848127

Message from the Organisers

Welcome to the first Workshop of Data-driven Approaches to Ancient Languages, marking a significant milestone within our interdisciplinary research project with the Database of Byzantine Book Epigrams. This workshop, collocated with the conference on Paratexts in Premodern Writing Cultures in Ghent, Belgium, underscores our commitment to fostering international collaboration and knowledge exchange in the dynamic fields of NLP and AI, as they intersect with ancient languages. We extend a warm invitation to all participants to immerse yourself in a productive exchange of ideas and insights throughout DAAL 2024.

This workshop has witnessed a notable surge in discussions surrounding data availability, scarcity and quality. In line with trends observed in recent conferences, large language models (LLMs) have dominated the technical discourse, highlighting their pivotal role in natural language processing. LLMs, however, are not always optimal for less-resourced languages, a label that is generally suited for ancient languages.

We extend our heartfelt appreciation to all authors who contributed to the programme of this workshop. Your dedication to sharing groundbreaking findings and innovations is the driving force behind the workshop's success. We are also immensely grateful to the program committee members who dedicated their time and expertise in reviewing submissions and steering the selection for the workshop. Additionally, we would like to express our gratitude to our invited speaker, Barbara McGillivray (King's College, London), for delivering an inspiring keynote speech.

In closing, we express our gratitude to the BOF21/GOA/028 fund, sponsored by Ghent University. Furthermore we would like to express our gratitude to the following instances for their generous funding: Department of Translation, Interpretation and Communication, VAIA - Flanders AI Academy, the Doctoral School, supported by the Flemish Government and the research group of Language and Translation Technology Team.

We wish you an enriching and enjoyable experience at DAAL 2024.

Colin Swaelens, Maxime Deforche, Ilse De Vos, Els Lefever

DAAL 2024 Organisers

Organizing Committee

Colin Swaelens, Language & Translation Technology Team, Ghent University, Belgium

Maxime Deforche, Database, Document and Content Management research group, Belgium

Ilse De Vos, VAIA - Flanders AI Academy, KU Leuven, Belgium

Els Lefever, Language & Translation Technology Team, Ghent University, Belgium

Scientific Committee

Luna De Bruyne, Antwerp University, Belgium

Katrien De Graef, Ghent University, Belgium

Guy De Tré, Ghent University, Beglium

Aaron Maladry, Ghent University, Belgium

Andreas Rhoby, Austrian Academy of Sciences, Austria

Pranaydeep Singh, Ghent University, Belgium

Gustav Ryberg Smidt, Ghent University, Belgium

Claudia Sode, University of Cologne, Germany

Toon Van Hal, Catholic University of Leuven, Belgium

Cynthia Van Hee, Ghent University, Belgium

Table of Content

"Nescio Carneades iste qui fuerit": Evaluation of Knowledge Bases for Named Entity Linking for Latin Texts.

Evelien de Graaf, Mark Depauw, Margherita Fantoli
KU Leuven
Faculty of Arts, Blijde Inkomststraat 21, 3000 Leuven, Belgium
{evelien.degraaf, mark.depauw, margherita.fantoli}@kuleuven.be

Abstract

In this paper, we discuss the feasibility and the challenges of Named Entity Linking (NEL) for Latin literature, focusing on the evaluation of a domain-specific Knowledge Base (KB). For this purpose, we discuss in detail two potential online datasets that could serve as KB: the People section of ToposText and *Paulys Realencyclopädie der classischen Altertumswissenschaft* (*RE*). From both of these resources, we construct a name dictionary to serve as KB for Candidate Entity Generation for NEL on Latin literature. The Candidate Generation relies first on partial and fuzzy matching, complemented with an extra exact match layer for multi-token entities. The coverage and reliability of the resources are evaluated quantitatively and qualitatively on two manually annotated texts, Tacitus *Historiae* and the *History* of Ammianus Marcellinus. This evaluation demonstrates that the *RE* is more suitable to function as basis for a KB for NEL on Latin. In addition, the qualitative analysis of the Candidate Generation method shows that NEL employing the *RE* is feasible and allows us to clearly define the remaining areas of improvement.

Keywords: Named Entity Linking, Latin, Knowledge Base, Knowledge Graph, Candidate Entity Generation, Fuzzy String Matching

1. Introduction

Named Entity Linking (NEL) or Disambiguation, often considered a sub-problem of Entity Linking (EL), focuses on the unambiguous identification of entities in text. Shen et al. (2015) define the task of NEL as given a set of entities E contained in a Knowledge Base (KB) or Graph (KG) and a collection of texts with a set of named entities M, entity linking aims to map each mention $m \in M$ to the corresponding entity $e \in E$. The task is commonly divided into three modules:

1. **Candidate Entity Generation**: generate all entities (E_m) from the KB that potentially match entity m.
2. **Candidate Entity Ranking**: given E_m, leverage predefined evidence to return the entity e that is most likely to be referenced by mention m.
3. **Unlinkable Mention Prediction**: a link to entity $e \in E$ will not be possible for each mention m, these are commonly labelled as NIL.

The most commonly used KBs for NEL are Wikipedia and Wikidata (Oliveira et al., 2021), even for linking Historical data such as, for example, newspapers and classic commentaries within the context of the HIPE shared task (Ehrmann et al., 2022). For ancient Greek and Latin, NEL is currently primarily manually executed on individual texts or small corpora as no fully automated pipeline has been published for this task. As will be detailed in Section 2, entities in these languages are manually or semi-automatically linked to diverse online resources but not always fully disambiguated within this process. These type of corpora are of limited use as many individuals might be grouped together under one identifier. An automatic approach to ancient Greek and Latin will allow for the linking of individuals within large corpora that opens up the way for new computational approaches to these individuals, such as Social Network Analysis.

With this paper, we aim to contribute to the development of NEL for Latin: first, we offer a detailed evaluation of two potential KBs for the linking of people in Latin literature; second, we evaluate the feasibility and challenges of a specific method for the first step of NEL, Candidate Generation, for persons in Latin literature; third, we publish a small set of gold data of disambiguated individuals.[1]

We start with an overview of related work on NEL in the domain of Classics in Section 2. In Section 3, we discuss two options for the KB and the process involved in the preparation of these resources for use in NEL, ToposText[2] in Section 3.1 and *Paulys Realencyclopädie der classischen Altertumswissenschaft* (*RE*)[3] in Section 3.2. In 3.3 we describe the creation of the gold standard used to evaluate the method. In section 4, we discuss how the KBs, transformed into name dictionaries,

[1] Data and code available at `https://github.com/evelien-degraaf/Named-Entity-Linking-Latin-DAAL-2024`.

[2] `https://topostext.org/`.

[3] Online available on Wikisource.

are employed for Candidate Entity Generation for Latin literature. The results are detailed in 5 and analyzed in 6.

2. Related work

Projects related to NEL in the domain of Classics primarily concern manual annotation of primary sources, translations and commentaries. Several of these projects also contain linked or disambiguated entities in varying levels of detail.

Several (online) corpora that focus on a specific type of text or subject contain **annotation** on disambiguated or linked entities. The *Latin Text Archive* (*LTA*)[4] and the *Patristic Text Archive* (*PTA*)[5] from the *Berlin-Brandenburg Academy of Sciences and Humanities* respectively contain Latin texts and Christian texts from antiquity till the Middle ages and annotations for people and places. The LTA does not provide any disambiguation beyond lemmas in the Frankfurt Latin Lexicon; the PTA does by providing project-specific URIs for all persons and links to STEPBible identifiers for biblical persons. The STEPBible project[6] itself contains annotations for people in the Greek Bible and provides an exhaustive disambiguated overview of all proper names in the Bible with URIs.[7] Another text annotated with disambiguated entities is the *Odyssey* by Josh Kemp.[8] He linked the majority of persons in this text to corresponding Wikipedia pages, except for minor characters that do not have a unique Wikipedia entry (Kemp, 2021). Furthermore, the Greek Fragmentary Tragedians Online also contains disambiguated annotations for persons with links to VIAF for authors and to Wikidata for other historical persons.[9] Currently, these annotations are only available for translation and commentaries not for ancient Greek (Antonopoulos et al., 2023). Last, another relevant project is the work of Christian Rollinger, who disambiguated individuals in the works of Cicero to enable the use of Social Network Analysis to study relations of *amicitia* in the Roman republic (Rollinger, 2014). The disambiguated individuals have where possible, been linked to the *RE*. Sadly, the data from this study is not openly available in digital format, which limits its usefulness for further use for this paper.

An example of specifically **semi-automatic** applications of NEL, are the projects of Monica Berti. For example, the new LAGL project aims to offer "a knowledge base for linking entity mentions to a structured vocabulary for ancient Greek authors and works that can be used to annotate other significant texts" (Berti, 2023, Project Content). The named entities within the the Digital Athenaeus project[10] are linked to many different resources such as the dictionary tool *Logeion*, the *Lexicon of Greek Personal Names* (*LGPN*), the gazetteer of places for the ancient world,*Pleiades*, and several others, but no further disambiguation is offered (Berti, 2021, pp. 402–5). Links exist at the name level, but no further disambiguation was performed. For example, selecting 'ΑΓΑΜΕΜΝΟΝΙ [AGAMEMNONI]' in the Named Entity Digger leads to the lemma Ἀγαμέμνων [Agamemnon] that links to several resources such as a lemma in *Logeion* and in the *LGPN*. The link to the personal name in *LGPN* then points to a record of the name Ἀγαμέμνων, which records ten different individuals with this name. These annotations and links were created in a semi-automatic manner and will in the future also include links to Wikidata for disambiguated persons (Berti, 2021, pp.398–414). Such links have already been established in an associated project, Digital Harpocration, for authors and works.[11] The diversity in these annotated corpora on choice of linked resources and level of disambiguation limits the interoperability and re-usability for fully automated NEL.

Fully automated NEL in the domain of Classics has only been attempted as part of the HIPE 2022 shared task (Ehrmann et al., 2022) as it included NEL on Classical Commentaries as contained in the *Ajax Multi-Commentary* project.[12] This project contains commentaries in English, German, and French with annotation linked to Wikidata IDs containing in total 7,482 mentions of which 1.45% marked as NIL. The authors describe the low percentage of NIL as "not at all surprising considering that commentaries mention mostly mythological figures, scholars of the past and literary works" (Ehrmann et al., 2022, p. 431). The participating teams score high on Named Entity Recognition (NER) on this dataset, but quite low on EL. First, this project demonstrates that the majority of entities in the AjMC project exist on Wikidata, but also that this is mainly due to the subject matter of the project. Second, it illustrates that even if the NER is accurate, EL can remain a challenge.

[4] https://lta.bbaw.de/.

[5] https://pta.bbaw.de/en/.

[6] https://www.stepbible.org/.

[7] Overview available on Github.

[8] Available from the Beyond Translation project.

[9] https://fragtrag1.upatras.gr/exist/apps/fragtrag/index.html.

[10] https://www.digitalathenaeus.org/.

[11] E.g. the links to Wikidata for author Callimachus and the works *On Contests* and *Hypomnemata* on the same page.

[12] https://mromanello.github.io/ajax-multi-commentary/.

3. Data

Currently, no Classics-specific KB exists that can be employed for NEL. Many different separate resources do exist, such as the *LGPN* and *Pleiades*. Other online available resources that collect people or names of entities are projects such as MANTO for mythological persons,[13] *Prosopographia Imperii Romani* (*PIR*) for persons alive during the Roman Empire,[14] *Digital Prosopography of the Roman Republic* (*DPRR*) for person alive during the Roman Republic,[15] and *Trismegistos* (TM) for a wide array of attestations of named entities.[16] However, many of these have a specific topical, chronological or geographical focus that makes them unfit to form a basis for a KB that can be used to identify all persons mentioned in ancient Greek and Latin literature. Therefore, we propose two different resources that could potentially fulfil this function. In Section 3.1, we discuss ToposText and the pre-processing of its People dataset (from now on TTP) and in Section 3.2 the *RE*. Last, in Section 3.3, we address the annotation process involved in preparing a corpus for NEL.

3.1. *ToposText*

For a first KB, we propose leveraging the names of people available in ToposText.[17] The ToposText project aims to make available an indexed collection of English translations of (primarily) Greek and Latin texts. In total, 809 texts have been annotated identifying people and places. The texts range from the Homeric poems all through to the 14th century CE, with a focus on texts written before the 3rd century CE. Besides covering a large period, the texts are also of a wide variety of genres, containing mostly non-documentary texts from the genres geography, history, nature, philosophy, reference, and "myth-literature".[18] The export of people from the online data results in 15,694 entries, all with a unique ToposText-People-ID. Additional information is present for some entries in the form of a description, a period, a Wikipedia link, a Wikidata ID, and a "Gender/Type".

To enable the use of TTP, we manually checked the attestations for 2,882 entries in the local instance. More precisely, based on the Genders/Types, all entries labelled as "Female" that occur ≥ 30 times have been annotated, and for "Male" those that occur between 50 and 1,000 times, while for "Religious" those that occur ≥ 3. For the remaining Genders/Types, those entries that occur ≥ 10 times have been annotated. No entities that have the Gender/Type "None" (4,391 in total) have been annotated. The annotation aimed at assessing how many entries in TTP refer to one individual person and whether the entries correspond to the individual identified with the associated Wikidata ID. Table 1 gives an overview of the annotation results recording the division of Genders/Types in TTP (Total), the number of annotated entities per Gender/Type (Annotated), and the number of actual individuals per Gender/Type (Identified Individual People). The entries recorded in this last column, 725 in total, have been exported to a separate dataset containing only those entries that attest a single individual.

The evaluation demonstrated that many of the Gender-/Type-labels are in fact categories that already clarify that the entity is not a person (e.g. "Datable event"). Second, the evaluation also brought to light some inconsistencies in the assignment of these "Gender/Type"-categories: for example, in the category "Animal" we find Epicureanism, a school of philosophy (TTP-ID: 9329), and Sappho, one of the most well-known female authors from antiquity (TTP-ID: 483), is labelled "Male". Furthermore, only ca. half of the "Female" and "Male" entries in TTP are individual people, while the rest contain annotations that refer to multiple individuals carrying the same name. For example, "Theodosius of Bithynia" (TTP-ID: 715; Wikidata ID: Q1266186) contains attestations that refer not only to this individual but amongst others also to the emperor Theodosius I (e.g. pseudo-Aurelius Victor, *Epitome de Caesaribus*) and II (e.g. Evagrius Scholasticus, *Ecclesiastical History*),[19] the Roman general known as Count Theodosius (Ammianus Marcellinus, *History*), and a youth from Thrace named Theodosius (Procopius, *Secret History*). Last, we observed several duplicates with different TTP-IDs and Wikidata IDs. For example, the god Pluto appears twice: once as "Hades (s. of Cronus), Roman Pluto" (TTP-ID: 10685; Wikidata ID: Q41410) and once as "Plouton, Pluto, god of the dead cf. Hades" (TTP-ID: 45; Wikidata ID: Q152262). These issues will influence the reliability of the identified individuals with NEL, as IDs do not unambiguously identify individuals.

Both the disambiguated individuals described above and the full TTP are used in Section 5 to evaluate TTP's coverage as a KB for NEL on Latin literature.

[13] https://manto.unh.edu/viewer.p/60/2616/scenario/1/geo/.

[14] https://pir.bbaw.de/.

[15] https://romanrepublic.ac.uk/.

[16] https://trismegistos.org.

[17] https://topostext.org/.

[18] This genre contains a very diverse array of texts ranging from many types of poetry (e.g. epic, hymns, elegy etc.) to philosophical treatises.

[19] Both emperors have in their own entry in TTP that is linked to the correct Wikidata ID: I = TTP-ID 15379 and II = TTP-ID 15380.

Gender/Type	Total	Annotated	Identified Individual people
Animal	62	62	0
Astronomic	22	22	0
Datable event	804	804	0
Ethnic	239	176	0
Female	1,448	228	138
Group	65	31	0
Male	7,941	1,280	528
Other	160	106	11
Place	288	85	2
Religious	206	73	46
Written Work	68	15	0
None	4,391	0	0

Table 1: Count per Gender/Type in TTP.

3.1.1. Transformation of *TTP*

From this data, we constructed name dictionaries following the ideas of Name Dictionary-based approaches as described by Shen et al. (2015, p.449). The dictionaries contain surface forms automatically extracted from associated Wikidata IDs that are recorded either as labels or aliases that have the language attribute "la". When there is no Wikidata ID associated with the ToposText entity, we use an automatically cleaned version of the ToposText "Header" as surface form: everything between brackets is deleted from the form and surface forms are split on ",". For example, "Canthus (s. of Canethus)" (TTP-ID: 3718) has no Wikidata ID associated with it. As adding "Canthus (s. of Canethus)" as surface form would complicate string matching, everything between brackets has been scraped off, resulting in the surface form "Canthus". In addition, for every name that contains a *v* or a *j* an alternative spelling is added as a potential surface form with *u* or *i*. For each surface form, all potential TTP-IDs are recorded.

3.2. *Paulys Realencyclopädie*

A second potential KB is proposed based on the *RE*. The *RE* is an encyclopedia meant to give a comprehensive overview of antiquity up until the time of Cassiodorus and Justinian aiming to deal with “all events and names of people of some importance” (Classen, 2010, p.4). The approach applied here to the *RE* has been made possible due to the ongoing effort to establish an online Wikisource edition of the original. This open edition allows unprecedented access to the *RE*, facilitating easy search through simple queries. For all entries in the physical edition, the online edition includes a web page, a short summarizing description (*Kurztext*), links to the entry's pages in the physical edition, and for public domain entries, the complete text from the physical edition (*Volltext*). Completed entries contain, where possible, links to the corresponding entry on Wikipedia and Wikidata. In the *Volltext* references to other entities in the *RE* are interlinked to either their corresponding web page or position in the register, which is complete and contains all *RE* keywords (*Stichwörter*).

Several aspects both of the original creation of the *RE* and of its digitisation influence the reliability of its potential as KB. First, in its original creation selection was necessary. Classen (2010) comprehensively summarises these omissions and inclusions:

- Entries can be hard to find due to unclear choices in *Stichwörter* and in the language of the *Stichwörter*;
- Some subjects receive more attention than one would expect or than needed and vice-versa, or are, according to Classen (2010), completely unnecessary;
- There are duplicates among the entries that are dealt with differently by the different editors. In addition, there is overlap between information contained within different entries;
- Some entries are not up-to-date anymore;
- Entries are missing.

Second, with the digitization of the *RE*, copyright became a prominent problem. The online version of the *RE* has to follow the German Wikisource guidelines to only publish articles in the public domain. These are articles written by an author who has been dead for over 70 years and articles that are too short to enjoy author copyright such as references and one-sentence articles.[20] For articles

[20] A full description of these guidelines can be found here on Wikipedia. Note that the scanned version of the books is in the public domain for all but the most recent entries.

that are not public domain yet, a placeholder is created and added to the register. Their goal is to create "98'717 Artikel +13'958 Nachträge + Kapitel (13'702 + 251 = 13'953), also 112'675 Elemente" and as of 05-01-2024, they report that 63.69% of articles have been created.[21]

3.2.1. Transformation of *RE*

The register of *Stichwörter* was used to create a local instance of the *RE* for further manual evaluation and annotation. This contains a total of 98,575 articles from all volumes including the supplements. As the *Kurztext* is available for almost all articles, we first exploited this to assign entries to categories such as 'god', 'animal', 'place' or 'person' by checking for the presence of keywords. In the second instance, we checked all entries manually. This resulted in a database of 48,720 entries identified as persons. As an alphabetic encyclopedia, the RE uses a single name to classify the entries, usually marked in bold. In the Wikisource version, only this element is provided as a title. Other personal names are given in the *Kurztext* (where available), but often in a non-standardized order. We restored the complete name where possible and split it up into its constituent elements. Each of these was linked to the corresponding TM name variant and name (TM NamVar and TM Nam ID respectively).[22]. Furthermore, we extracted any dates available from the *Kurztext* or else from the *Volltext*.

From these entries, we constructed a name dictionary for NEL based on the complete name and the name components.[23] The complete name and separate components are recorded as potential surface forms and for every name that contains a *v* or a *j* an alternative spelling is added as surface form with *u* or *i*. Furthermore, incomplete name components – those that contain "..." – have been excluded as these are deemed irrelevant to the study of literary texts as foreseen here. Last, any entries that contain multiple different surface forms, separated by a "/" in the local instance of the *RE*, have been split up automatically into potential surface forms. For example, the entry 'Abdemon / Audymon' becomes potential surface forms 'Abdemon' and 'Audymon'.

3.3. Gold Annotated Data

For evaluation of Candidate Generation using the data from TTP and *RE*, we use a text from the LASLA Latin corpus[24] and one from the LTA project. The LASLA corpus contains a diverse range of Latin texts composed of 1,738,435 tokens, belonging to 130 Latin literary texts by 21 authors ranging from the 2nd century BCE to the 2nd century CE. Each token has been manually lemmatized. The corpus has already been employed for testing NER tools and the results of the best functioning model are available online together with a manually annotated part of the corpus used as gold standard (Beersmans et al., 2023).[25]

We furthered the annotation of the LASLA NER gold standard Tacitus *Historiae* 1 for NEL. All entities marked as persons, ie. with the annotation PERS, were annotated to include disambiguation of persons by linking to a TTP-ID and a *RE*-ID. Of the 804 person tokens, 13 were not annotated with either a *RE*-ID or TTP because they belonged to categories described in Table 2. The 791 annotated tokens represent a total of 634 entities of which 164 are multi-token entities and 470 are single-token.

The annotation with a *RE*-ID was done manually by searching the online Wikisource edition for a match to the person mentioned in the text. In total, only three entities could not be found in the *RE*: *Aegialus* (1.37), *Crispina* (1.47), *Petrianus* (1.70). Another issue encountered during the annotation, already identified in Section 3.2, is that several tokens, 9 in total, had to be annotated with multiple IDs as these IDs are duplicates of the same individual.[26]

The annotation with a TTP-ID was slightly more complex. The project offers its own entity annotation in a translation.[27] This annotation was copied manually to the Latin. For multi-token entities, ToposText either annotated all tokens with a separate ID or only one of the tokens. Of the 791 entity tokens, 133 have not been annotated by ToposText. In addition, 58 tokens have been annotated with an ID that is not a match for the individual referenced with the token. One such a case is the annotation of all occurrences of the name *Ualens* (e.g. 1.7 *Fabius Ualens* and 1.56 *Donatius Ualens*) as referring to TTP-ID 466 identified as "Eastern Roman Emperor from 364 to 378 (328-378)", which is incorrect. We created a cleaned version of this annotation to see if this would enhance NEL (from now on "distilled annotation"). First, all tokens in the multi-token entities are annotated with the same ID. Second, entities that appear multiple times in the text are identified with the same ID throughout the text. Third, each ID only appears once and last, the individual referenced with the ID matches the

[21] https://elexikon.ch/?Typ=RE&Text=RE-0_Stand.htm.

[22] A search tool facilitating access to the RE based on this work is now available at https://www.trismegistos.org/real.

[23] The full *RE* name dictionary is available on GitHub.

[24] https://www.lasla.uliege.be.

[25] Available on Github.

[26] It is possible that more doubles are present in the *RE* that remained unnoticed during the annotation process.

[27] https://topostext.org/work/199.

Category	Tacitus	Ammianus	Examples
Gods / personifications	4	10	*Penatis / Fortuna*
Collective of persons	4	2	*Iuliorum Claudiorum / Gordianorum*
Generic / title	5	2	*imperator / princeps*

Table 2: Number of person entities not annotated per category in Tacitus and Ammianus.

individual in the text. In total, 116 entities remain unidentified because either it is missing from TTP (e.g. for *Caluia Crispinilla*, 1.73), or an entry exists under that name but is identified as another individual (see the *Ualens* example) or that name has already been used to identify another individual in the text. For example, for *Cadius Rufus* (1.77) only for *Rufus* an entry exists but that one has previously been used to identify *Cluuius Rufus* (1.8).

In addition, we manually annotated book XIV of the *History* of Ammianus Marcellinus to further evaluate the coverage of the *RE*.[28] The text edition was taken from the Latin Text Archive and contains automatically generated NER annotation.[29] In addition, the text has been lemmatized automatically.[30] We manually verified and corrected the NER annotation to follow the same standards as for Tacitus. Due to complications caused by the inclusion of terms such as *imperator* and *princeps* for Tacitus, we did not annotate these in Ammianus. In total, this text contained 236 person entity tokens, representing a total of 215 entities of which 19 are multi-token entities. These were again manually annotated with a *RE*-ID. 14 entity tokens were not annotated with a *RE*-ID or because they belonged to the categories described in 2. This text contained seven entities that could not be found in the *RE*: *Catena* (5.8), *Paulus* (5.6, 5.8 (2x), 5.9), *Sannio* (6.16), *Eubulus* (7.6), *Epigonus* (7.18), *Apollinaris* (7.19), *Gundomadus* (10.1). Multiple entries in the *RE* for one individual were observed for six tokens.

3.3.1. Inter Annotator Agreement

A section of both texts was annotated by two Classics experts and the disagreement was discussed in detail. After discussion, the annotators agreed on the annotation of all person entities. Some disagreement stemmed from differences in boundaries for entities identified by the NER annotation on Ammianus. However, the majority of disagreement stemmed from the difficulty of identifying the correct individual within the text without further contextual information. Difficulties arise in, for example, annotation of same-named individuals such as *Constantius* that is linked to either Constantius 3, Constantius 4, or Constantius 5 in Ammianus. Another example is selecting the correct *Ptolemaeus* out of 83 *Ptolemaios RE*-entries or *Iulius* out of 599 *RE*-entries to match occurrences of these names in Tacitus. We expect these issues will impact the feasibility of automated NEL overall and will thus remain open for further research.

4. Method

We focus specifically on the first step of NEL: generating a Candidate set from a KB that could be a potential match for the mention in context. In general, Candidate Generation employs the following approaches either separately or combined (Sevgili et al., 2022):

- **Surface form matching**: using approaches such as edit distance, n-grams, and normalization. Employed in isolation this method can be inaccurate as it does not take into account aliases and nicknames.
- **Surface form Expansion using aliases**: aliases, and nicknames can be expanded to create more accurate surface form matching for these by, for example, exploiting metadata of the knowledge base or synonym/antonym dictionaries.
- **Probability + expansion using aliases**: employs a pre-calculated prior probability using Wikipedia entity hyperlinks, CrossWikis, or another way to determine the "popularity prior".

Central to Candidate Generation is the problem of matching a surface form to a KB surface form. As discussed by Shen et al. (2015, pp.449, 452), two possible approaches are retrieving all the full matches for the mention or using partial matching followed up with different approaches that prioritize different aspects.

Our approach relies upon partial surface form fuzzy matching with expanded names and aliases being recorded in the name dictionaries, followed by an exact matching of multi-token entities. The name dictionaries contain different variants of potential surface forms and common abbreviations

[28] Based on the unsatisfactory results on Tacitus for TTP detailed in Section 5, we decided to continue only with the *RE*.

[29] Available on their website.

[30] Their website states "If possible, the lemmatization results have been checked and curated by latinists". It is unclear whether such a check took place for Ammiunus.

for names, such as M. for Marcus or C. for Caius. At this point, prior popularity is not taken into account. The matching pipeline consists of the following steps (for a visualisation of the pipeline, see Figure A in Appendix A):

1. Reconstruct entity surface forms based on B-PERS and I-PERS annotation and lemmas or tokens when no lemma is available.
2. Match surface forms to name dictionaries using RapidFuzz Fuzzy matching process `token_ratio`.[31] Pre-processing is set to default which trims whitespaces, ignores numbers, and lowercases the strings. In addition, it sorts and sets the tokens in a string before matching, therefore disregarding word order and repetition. We placed the score cutoff at 88 after experimentation and specified no limit on candidate suggestions.
3. For multi-token entities, limit the potential candidates to those that contain an exact match for all tokens of the surface form.

The details for the code used with the rationale behind the choices for the specific matching algorithm are available in a Jupyter Notebook available on GitHub.

5. Results

Tactitus *Historiae* 1 contains a total of 791 annotated person tokens. The predicted candidate lists employing the full TTP contain the correct candidate for 78.13% of the tokens when compared with ToposText's annotation. When using only the disambiguated individuals, the correct candidate is present only for 20.99% of the entities. When compared to the distilled annotation (described in Secton 3.3), the percentage is slightly higher for the full TTP, 79.77%, but lower for the individuals, at 19.34%. For the *RE* annotation, the correct candidate is present in the generated candidate list for 92.16% of the total entity tokens. The *History* of Ammianus contains 223 annotated person tokens. The correct candidate is present in the candidate list generated with the *RE* for 47.53% of the total tokens. Table 3 shows the number of entities that belong to the categories of 0 candidates proposed, 1 candidate proposed, between the 2 and 10 candidates proposed and more than 10 candidates proposed.

6. Discussion

The evaluation of ToposText indicates that this data is not suitable to fulfil the function of KB for Latin NEL for the following reasons. As concluded in Section 3.1, the reliability of any identification using the full dataset is impacted by the fact that many IDs do not refer to one individual. The results shown in Table 3 establish that using only the disambiguated persons results in the prediction of 0 candidates for the majority of the entities and is consequently not useful for NEL. Furthermore, fuzzy matching on a description rather than a standardized name is undesirable. Relying on Wikidata labels and aliases partly solves this difficulty, but for many entries, no Wikidata-ID is recorded. Besides, sometimes noisy data is present in the aliases, such as, for example, epigraphic aliases such as "CAIVS•IVLIVS•CAESAR•IV" for Iulius Caesar.[32] Thus, the results, in addition to the complications encountered in transferring ToposText's annotation of Tacitus, demonstrate that ToposText can not serve as a KB for Latin literature.

The *RE* covers the majority of entities present in the texts, as detailed in Section 3.3. However, a close analysis of the results further emphasises several of the limitations also identified in Section 3.3. First, missing entries were present in both texts: 3 entities in Tacitus and 6 in Ammianus could not be identified using the *RE*. Second, multiple entries were observed for 15 entities. Last, during the annotation entries were encountered that refer to multiple individuals. One example is Asiaticus 8a-e which contains 5 different individuals under the same entry. For future NEL applications, we aim to address these issues by editing the local instance of the *RE* to include new identifiers for missing entries and entries that are located within one *RE*-ID, and to merge several *RE*-IDs under one where multiple entries are present for one individual.

6.1. Evaluation of Method

Based on the results, several challenges and limitations of the method can be identified. Especially the low number of candidate lists that contain the right candidate for the entities in Ammianus offers valuable insights.

Several issues are related to the **first step** described in Section 4, reconstruction of the entities based on lemmas. This is illustrated clearly by one of the reasons for the low probability of correct candidate prediction on Ammianus: the text edition contains both missing lemmas and incorrect lemmas. Missing lemmas are observed for 10 tokens, where for 3 the lemma is "ProperName" and for 7 it is empty. For these cases, the token is used for fuzzy matching but as these often are in different cases than the lemma form, string matching only results in the correct candidate being present for three out of these ten cases. Incorrect lemmas are

[31] https://maxbachmann.github.io/RapidFuzz/Usage/fuzz.html.

[32] https://www.wikidata.org/wiki/Q1048.

	Tacitus			Ammianus
	Disambiguated TTP	**Full TTP**	***RE***	***RE***
0 candidates	563	29	3	22
1 candidate	203	134	126	15
Between 2-10 candidates	25	604	223	71
More than 10 candidates	0	24	439	115

Table 3: Length of candidate lists proposed for Tacitus using the three different name dictionaries and Ammianus using the *RE*.

cases such as *Galla*, *Galli*, or *Gallo* for forms of *Gallus* or *Constantio*, *Constans* or *Constantia* for forms of *Constantius*. This demonstrates that automatic candidate generation in this form is reliant on the presence of correct lemmatization, which will not be available for every text. A second issue caused by reliance upon lemmas for surface forms is names used as adjectives associated with an adjective lemma form such as forms of the lemma "uitellianus" (e.g. Tac. *Hist.* 1.51:*motus Uitelliani* or 1.75: *Uitellianis inpune*) or "neronianus" (Tac. *Hist.* 1.23 : *Neroniani comitatus*).

The **second step**, employing fuzzy string matching, has limitations when the name dictionary is not exhaustive for all name variants and spelling variations. This issue is exemplified by the entities *polyclitus* (Tac. *Hist.* 1.37: *Polycliti*) and *pacorus* (Tac. *Hist.* 1.40: *Pacorum*). These are linked to Polykleitos 5 and Pakoros 2 respectively. However, the correct entries do not appear in the candidate lists created with fuzzy matching as their surface form scores below the threshold score of 88. This issue furthers the problems caused by missing and incorrect lemmas in Ammianus as neither the token nor the incorrect lemmas will produce scores above the threshold score when fuzzy matching. The issue of missing name variants is exemplified in Ammianus by the surface forms *Caesar* and *Gallus*. Neither form is recorded as a potential name variant for main character Constantius 5, causing 41 entity tokens not to match. This particular issue is addressed by manually adding name variants to the local instance of the *RE*. A solution to the larger problem would be to expand the name dictionary with surface forms extracted from TM. For each name component, it is possible to exploit the link to TM Nam to extract all possible Latin NamVars to include different spelling variants. For example, *Polykleitos* is linked to TM Nam 5219 which has *Polycletus* (NamVar ID: 125169) and *Polyclitus* (NamVar ID: 131377) listed as Latin name variants. Recording both these variants as potential surface forms for *Polykleitos* enables linking based on fuzzy matching to the surface form *polyclitus*.

The **third step**, limiting candidate suggestions for multi-token entities based on exact matches, is overall beneficial for NEL. Only for a total of 10 out of 164 multi-token entities in Tacitus, this step eliminates the correct candidate from the candidate list. This is the case for *Fonteius Capito* (5x), *Tiberius Nero*, *Germanicus Uitellius*, *Plotius Firmus* (2x), and *Gaius Iulius*. In Ammianus, the correct candidate is not predicted for 12 out of 19 multi-tokens, 11 of which are not caused by the multi-token step. For these entities, the correct candidate is not predicted due to the issues identified in the first two steps: missing name variants in the name dictionaries (e.g. variations of *Gallus Caesar*), incorrect lemmas (e.g. *Alexandra Magnus* (11.22)) or spelling variations (e.g. *Nicator Seleucus* (8.5)). Only for one of the 12 incorrect multi-tokens, *Valerius Publicola*, did the multi-token step eliminate the correct candidate. For these entities, the exact match is not present in the complete name of the correct entities for diverse reasons.

Some issues require manual changes. For *Fonteius Capito*, a transcription error is present in the online entry in the *RE* where Fonteius 18 is recorded as "Fonteíus", with an acute accent on the "i". In the associated scanned page, the name is simply "Fonteius". In the case of *Tiberius Nero*, the problem is caused by the name *Nero*: the correct entity, Iulius 154, was listed under the complete name *Tiberius Iulius Caesar Augustus* with the pre-adoption name *Tiberius Claudius Nero* not recorded. In the case of *Germanicus Uitellius*, the match to Vitellius 7b is not made because *Germanicus* is not in the recorded complete name of *Aulus Uitellius*. These issues are addressed by manually editing the local copy of the *RE*.

The issue with *Plotius Firmus*, *Gaius Iulius*, and *Valerius Publicola* is related to the limitation of spelling variants of names discussed in Section 6. *Plotius Firmus* should match to Plotius 2, complete name *Plotios Firmus*, *Gaius Iulius* to Iulius 131, complete name *Caius Iulius Caesar*, *Valerius Publicola* to Valerius 302, complete name *P. Valerius Poplicola*. *Plotios* is not an exact match for *Plotius*, *Caius* not for *Gaius*, and *Poplicola* not for *Publicola*. This challenge could be addressed in multiple ways. The first option is changing the multi-token step to fuzzy matching as well instead of requiring an exact match for all tokens present in the entity. However, the requirement of an exact match also significantly shortens the predicted candidate lists. For example, *nymphidius sabinus* is to be linked to Nymphidius 5; before applying the multi-token step, the candidate list is 151 entries long, after, the number of potential candidates is reduced to just the correct one. For 49 out of 164 multi-token entities in Tacitus, this is the case. No such cases are observed in Ammianus. Another option would be to include more name variants in the name dictionary by exploiting established links to the TM database as detailed above.

7. Conclusion

Overall, this paper demonstrates that Candidate Generation employing the data from *RE* as KB for Latin is feasible. The coverage of the *RE* is significantly better than TTP as only 10 entities are unlinkable to the *RE*, whereas for TTP 116 entities could not be identified in Tacitus alone. Future challenges remain in issues related to incorrect lemmatization and spelling of potential surface forms. The analysis of the step of limiting multi-token entities to exact matches also demonstrated its effectiveness, as it causes problems only in a minority of cases and limits the potential candidates significantly in others. Future work will focus on enhancing the coverage of the name variants and aliases recorded in the name dictionary, testing the coverage of the *RE* for Ancient Greek literature, and evaluating potential methods for Candidate Entity Ranking.

8. Limitations

The already highlighted incompleteness of the name variants in the name dictionary is a clear limit of the current approach. We already identified the potential of exploiting TM NamVar to improve the coverage (6.1). Another potential limitation of the approach outlined in this paper is that we rely upon a simple method of matching the surface forms to the name dictionary. Further development employing a neural approach, for example using DeezyMatch (Hosseini et al., 2020), could improve the results of matching. Last, we only evaluated the coverage of the *RE* for two well-known and well-researched Latin texts: it remains to be seen how complete the *RE* is when trying to link entities in less well-known ancient literature.

9. Acknowledgements

We thank the anonymous reviewers for their constructive reviews. We would also like to thank the teams behind the digitisation of the *RE*, the *LTA*, and *LASLA* as without such efforts research like this would not be possible.

References

Andreas Antonopoulos, Stylianos Chronopoulos, Nikolaos Ntaliakouras, Panagiota Taktikou, Anastasia Psomiadou, and Iraklis Markelis. 2023. Developing a Database for the Greek Fragmentary Tragedians. *Digital Classics Online*, pages 15–29.

Marijke Beersmans, Evelien de Graaf, Tim Van de Cruys, and Margherita Fantoli. 2023. Training and Evaluation of Named Entity Recognition Models for Classical Latin. In *Proceedings of the Ancient Language Processing Workshop*, pages 1–12.

Monica Berti. 2021. *Digital Editions of Historical Fragmentary Texts*. Digital Classics Books. Propylaeum.

Monica Berti. 2023. Named Entity Recognition for a Text-Based Catalog of Ancient Greek Authors and Works. In *Digital Humanities 2023: Book of Abstracts*, page 557.

Carl Joachim Classen. 2010. «Vita brevis – ars longa»: Pauly's beginnings and Wissowa-Kroll-Ziegler's monumental achievement. *Eikasmós*, 21:423–437.

Maud Ehrmann, Matteo Romanello, Sven Najem-Meyer, Antoine Doucet, and Simon Clematide. 2022. Overview of HIPE-2022: Named Entity Recognition and Linking in Multilingual Historical Documents. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, volume 13390 of *Lecture Notes in Computer Science*, pages 423–446. Springer International Publishing.

Octavian-Eugen Ganea and Thomas Hofmann. 2017. Deep Joint Entity Disambiguation with Local Neural Attention.

Kasra Hosseini, Federico Nanni, and Mariona Coll Ardanuy. 2020. DeezyMatch: A Flexible Deep Learning Approach to Fuzzy String Matching. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 62–69. Association for Computational Linguistics.

Josh Kemp. 2021. Beyond Translation: Building Better Greek Scholars.

Italo L. Oliveira, Renato Fileto, René Speck, Luís P. F. Garcia, Diego Moussallem, and Jens Lehmann. 2021. Towards Holistic Entity Linking: Survey and Directions. *Information Systems*, 95.

Marco Passarotti, Francesco Mambrini, Greta Franzini, Flavio Massimiliano Cecchini, Eleonora

Litta, Giovanni Moretti, Paolo Ruffolo, and Rachele Sprugnoli. 2020. Interlinking through lemmas. the lexical collection of the lila knowledge base of linguistic resources for latin. *Studi e Saggi Linguistici*, 58(1):177–212.

Maria Pershina, Yifan He, and Ralph Grishman. 2015. Personalized Page Rank for Named Entity Disambiguation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 238–243. Association for Computational Linguistics.

Christian Rollinger. 2014. *Amicitia Sanctissime Colenda. Freundschaft Und Soziale Netzwerke in Der Späten Republik*. Verlag Antike.

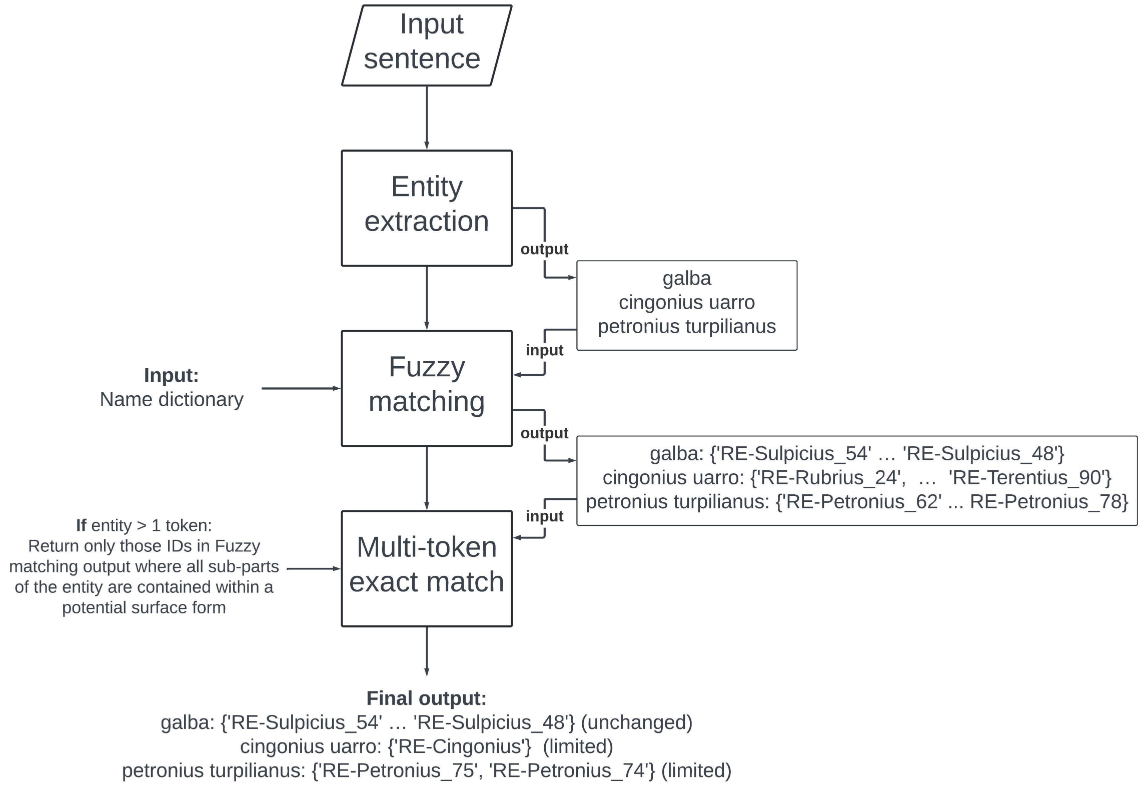
Ozge Sevgili, Artem Shelmanov, Mikhail Arkhipov, Alexander Panchenko, and Chris Biemann. 2022. Neural Entity Linking: A Survey of Models Based on Deep Learning. *Semantic Web*, 13(3):527–570.

Wei Shen, Jianyong Wang, and Jiawei Han. 2015. Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *IEEE Transactions on Knowledge and Data Engineering*, 27:443–460.

Example Sentence (# sent_id = TacHist1-Q-05-26)

NER Annotation: B-PER B-PER I-PER B-PER I-PER
tardum ***Galbae*** *iter et cruentum interfectis* ***Cingonio Uarrone*** *consule designato et* ***Petronio Turpiliano*** consulari
Lemma tardus galba iter et cruentus interficio cingonius uarro consul designo et petronius turpilianus consularis

Translation: Galba’s journey was slow and bloody after Cingonius Varrus, the consul designate, and Petronius Turpilianus, a former consul, had been killed


Input sentence
Entity extraction
output
galba
cingonius uarro
petronius turpilianus
input
Input:
Name dictionary
Fuzzy matching
output
galba: {'RE-Sulpicius_54' … 'RE-Sulpicius_48'}
cingonius uarro: {'RE-Rubrius_24', … 'RE-Terentius_90'}
petronius turpilianus: {'RE-Petronius_62' ... RE-Petronius_78}
input
If entity > 1 token:
Return only those IDs in Fuzzy matching output where all sub-parts of the entity are contained within a potential surface form
Multi-token exact match
Final output:
galba: {'RE-Sulpicius_54' … 'RE-Sulpicius_48'} (unchanged)
cingonius uarro: {'RE-Cingonius'} (limited)
petronius turpilianus: {'RE-Petronius_75', 'RE-Petronius_74'} (limited)

A. Illustration NEL pipeline

Automatic Generation of Ancient Greek Word Forms: a Mixed, Corpus-Based and Rule-Based Approach

Alek Keersmaekers
University of Leuven
Blijde-Inkomststraat 21
alek.keersmaekers@kuleuven.be

Abstract

This paper presents a system that aims to automatically generate Greek word forms, based on their lemma and morphological attributes. Currently barely any systems are capable of this, and those that exist are limited in scope. In contrast, by exploiting both a large corpus and a rule-based tool, viz. Morpheus, it is possible to achieve acceptable results for a large number of Greek varieties. Several applications are shown, including a more statistical approach to morphology learning, the ability to 'mask' specific morphological properties of Greek words for annotation purposes, and an experimental approach to Greek dialect identification. Some limitations include data sparsity for less-attested language varieties, a heavy reliance on what is included in Morpheus, and a rather binary approach to what constitutes the language variety of a text.

Keywords: Ancient Greek, morphology, word form generation, language variation

1. Introduction

Ancient Greek inflectional morphology has several layers of complexity: depending on the part-of-speech, Ancient Greek words can be inflected for person, number, tense/aspect, mood, voice, gender, case, and degree. This may involve both stem changes as well as affixation, while the stems and affixes being chosen may vary widely among dialects as well as diachronically. Various tools have been developed to automatically analyze inflected Ancient Greek forms, the most widely used one being Morpheus (Crane, 1991). However, tools that work in the opposite direction, viz. the *generation* of Ancient Greek word forms, are much more sparse (see Section 2). The aim of this paper is therefore to fill this current gap and present the ongoing development of a tool that can generate a given word form, starting from a lemma (e.g. λέγω 'to say') and a morphological tag (e.g. verb, 1 singular aorist indicative active).

There are multiple ways to do so, each with their own advantages and disadvantages. One option is to do this fully rule-based, on the basis of lists of words, associated with their possible stems and inflectional class(es), and endings associated with these inflectional classes. However, apart from the massive effort that compiling such lists would require if the system needs to be as exhaustive as possible, one also has to be very careful not to overgenerate (i.e. generate forms that one would not expect for a given time period or dialect, or are very infrequent) or undergenerate (i.e. not be able to generate those forms that would be appropriate for this period or dialect), given the large variety of the Ancient Greek corpus. Another option is a corpus-based approach, where the most frequent form for a given lemma/morphology combination is retrieved from an Ancient Greek (sub)corpus. This would eliminate the problem of overgeneration, given that the selection could be based on a subcorpus representing the specific variety that one would be interested in. However, it would quickly lead to data sparsity, given the highly inflectional nature of Ancient Greek. An Ancient Greek verb for example, has more than 200 possible forms depending on its inflection: to fully model the inflectional morphology of just 1,000 verbs, the absolute minimum corpus size would already be more than 200,0000 word forms. Finally, one could try a machine learning approach: for example, Ancient Greek word forms could be generated character by character based on their lemma and part-of-speech using a seq2seq approach (see e.g. Kanerva, Ginter, and Salakoski, 2021 for lemmatization, where lemmas are generated based on word forms and part-of-speech tags: in principle one could also learn from data to go in the other direction, i.e. generate word forms based on lemmas and part-of-speech). While some researchers have applied such an approach to Ancient Greek already, it was not very successful (see Section 2). Although this might be related to the fact that these approaches were carried out without taking any domain knowledge about Ancient Greek morphology into account, it is important to point out that the morphology of inflectional languages is difficult to process computationally (e.g. Hajic, 2000). Researchers who have worked on other tasks related to Ancient Greek morphology, such as morphological tagging (Keersmaekers, 2020) and lemmatization (Vatri and McGillivray, 2020), have also pointed out that approaches fully based on machine learning fare poorly – hence for such a 'reverse lemmatization' approach the same problems would likely arise.

To address these problems, this paper will present a mixed approach, where the advantages of a rule-based and a corpus-based approach are combined. This system enables users to generate Ancient Greek word forms in several language varieties: while the generated forms are based on corpus evidence (i.e. the system aims to generate forms that would be expected in a subcorpus representing a specific language variety), due to the interaction with a rule-based system (see Section 3) many more forms than the ones that are strictly attested in this subcorpus can be generated. Various use cases are discussed, including applications in didactics, linguistic annotation and dialect identification. After briefly discussing related work (Section 2), Section 3 will explain how this system works, and Section 4 will show some general results. Finally, Section 5 will show some possible applications of this system, and the main open challenges will be addressed in Section 6.

2. Related work

Projects related to the generation of Ancient Greek word forms are rather scarce: the only project I could find was a Python library developed by James Tauber (greek-inflexion), a rule-based approach based on lists of stems and endings as well as rules to combine them. The stems are taken from a number of learner resources (Tauber, 2016), viz. Louise Pratt's *The Essentials of Greek Grammar*, Helma Dik's *Nifty Greek Handouts*, and Keller and Russell's *Learn to Read Greek*. The scope of these learner resources is very limited, however (they contain just 19, 10 and 33 verb lemmas respectively, several of which overlap), and limited to the Attic Greek dialect. As a consequence, the number of word forms that this tool can generate is also extremely small.

Another system developed to handle Ancient Greek morphology is Morpheus (Crane, 1991): as mentioned in the introduction of this paper, its primary function is to analyze rather than generate Ancient Greek word forms, however. Morpheus starts from a list of lemmas, associated to one or multiple stems, which in turn are associated to one or multiple morphological paradigms, as well as a list of endings associated to these paradigms. Based on these lists as well as various rules to combine the stems with the endings, Morpheus generates a large database of inflected Ancient Greek forms. The actual analysis involves a lookup in this database. This means that Morpheus can theoretically be used as a generator instead of an analyzer. However, in such a case Morpheus would simply generate all possible forms for a given lemma and morphology combination, without distinguishing between frequency or appropriateness for a specific language variety (viz. the problem of 'overgeneration' discussed in the introduction of this paper). Nevertheless, given the extensiveness of this tool, it will provide a solid base for the system described here, while corpus evidence will be used to address overgeneration, as will be discussed in the next section.

Finally, it is worthy to note that the generation of inflected word forms is a task that has been tackled for several other languages already. In particular, SIGMORPHON (*Special Interest Group on Computational Morphology and Phonology*) has organized multiple shared tasks involving this topic (Cotterell et al., 2016; 2017; 2018; McCarthy et al., 2019; Vylomova et al., 2020; Pimentel et al., 2021; Kodner et al., 2022; Goldman et al., 2023). Since the focus is on multiple languages, the systems submitted for this task are typically based on machine learning (rather than a rule-based approach tailored to a specific language, as in this paper). For example, the best performing system in Goldman et al. (2023) was a transformer-based encoder-decoder model (Canby & Hockenmaier 2023). Nevertheless, in some cases (e.g. Beemer et al., 2020; Merzhevich et al., 2022) rule-based systems were submitted, which typically generate better-results than machine-learning systems, but require much more human effort (Kodner et al., 2022: 184).

Interestingly, in the most recent iteration of this task (Goldman et al. 2023) Ancient Greek was included as well. As for this paper, the aim was to generate a word form from a lemma and a list of morphological features: the data for Ancient Greek was based on inflection tables from Wikisource (Kirov et al. 2016). Even though this dataset only includes the classical Attic language variety and only consists of nouns and adjectives, Ancient Greek was still one of the languages the submitted systems struggled the most with, with the best system achieving an accuracy of only 56% (the second worst of 26 languages, with only Navajo performing slightly worse at 55.6%, while most other languages were in the 80-100% accuracy range).

3. System description

As described in the introduction in this paper, this paper will describe a mixed, rule-based and corpus-based system. In order to obtain substantial frequency information, a large corpus of Ancient Greek is necessary. This system is based on GLAUx (Keersmaekers, 2021), the largest openly available Ancient Greek corpus (8th century BC-4th century AD) available so far, containing more than 25 million tokens.

The most simple way to leverage corpus information to generate Ancient Greek word forms would be to simply retrieve all forms that occur in the corpus for a given lemma-morphology combination. However, as mentioned in the introduction of this corpus, this would quickly lead to data sparsity. For example, the verb ποιέω ('make', 'do') is a very frequent verb in the GLAUx corpus, occurring 68,213 times. However, even such a frequent verb has possible inflectional forms that are never attested in GLAUx: for example, the aorist passive singular feminine dative participle never occurs. This is simply a product of the many combinatory possibilities for the various morphological features (as well as frequency discrepancies among these features): other aorist passive participles of ποιέω do occur in GLAUx (81 in total), but simply never in the singular feminine dative case by accident. Obviously, the number of accidentally unattested forms will also grow to a large extent if only a specific subcorpus of GLAUx is selected (e.g. the subcorpus of writers who use the Ionic language variety).

However, the aorist passive singular feminine dative participles of other verbs that are inflected similarly to ποιέω do occur in GLAUx: one example is στερηθείσῃ of the verb στερέω ('deprive'), which is entirely analogically formed to the equivalent participle of ποιέω in the Attic dialect, viz. ποιηθείσῃ. In other words, if we can capture the fact that a) this specific participle of εω-verbs ends in -είσῃ and b) the passive aorist stem of ποιέω is ποιηθ-, it would be possible to generate the form ποιηθείσῃ without it occurring once in the corpus.

To capture this information, I started from Morpheus, which can return information about the morphological make-up of a given form. Taking the στερέω/ποιέω case as an example, for the participle στερηθείσῃ, Morpheus records the following information (apart from all the inflectional features such as aorist, singular etc.): {inflectional class: aor_pass, prefix: none, stem: στερηθ, augment: none, ending: εισῃ}. Similarly, other passive aorists of ποιέω would be assigned to the inflectional class aor_pass as well and receive the stem ποιηθ, so this is exactly the information we need to construct the form ποιηθείσῃ.

Concretely, I analyzed the full GLAUx corpus using Morpheus.[1] Morpheus was certainly not able to analyze all forms occurring in this corpus: of the 667,894 form types present in GLAUx, 84,617 (13%) were unrecognized. However, since most of these types have a rather low token frequency (350,176/21,638,098, or 1.6% tokens of GLAUx are unrecognized), this is not such a large problem as it might seem at first sight. Nevertheless, since several of the unrecognized forms are dialectical, the implications will be discussed in the conclusion of this paper.

After doing so, for each lemma present in GLAUx frequency information was collected in a pipeline process as follows (a step which roughly corresponds to the training process of traditional machine learning):

1. **Stem**: for verbs: how often does each stem occur for a given principal part of a given lemma?[2] For nouns/adjectives: how often does each stem occur of a given lemma? For example, in Attic prose texts, the stem ποιηθ- occurs 24 times for the aorist and future passive of the verb ποιέω.
2. **Inflection**: for verbs: how often is a particular inflectional paradigm used for a given stem of a given principal part of a given lemma? For nouns/adjectives: how often is a particular inflectional paradigm used for a given stem of a given lemma? The ‘inflectional paradigm’ refers to the set of endings an Ancient Greek form can take – while this is typically highly dependent on the stem, some stems may be assigned to multiple inflectional paradigms (see Figure 1, where the stem εἰπ- may be combined both with the inflectional paradigm ‘aor1’ and ‘aor2’). For example, in Attic prose texts, the inflectional paradigm ‘aor_pass’ is used 24 times for the stem ποιηθ- for the aorist and future passive of the verb ποιέω.
3. **Ending**: how often is a particular ending used for a given inflectional paradigm and a given list of morphological features? For example, for the paradigm ‘aor_pass’ and the morphological features ‘verb, aorist passive singular feminine dative participle’ the ending -είσῃ occurs 13 times.

This information is then dynamically combined, as illustrated in Figures 1 and 2.

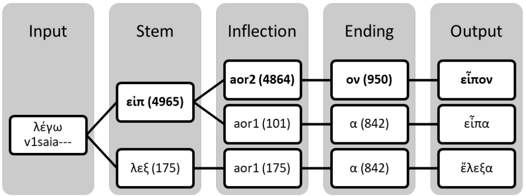


Figure 2: Example of verb generation.

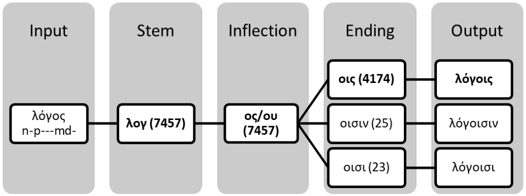


Figure 1: Example of noun generation.

This process is non-deterministic: for a given form, the system can generate multiple forms, as illustrated on the figures above. However, it can be made deterministic by making a selection based on frequency: in this case, it greedily selects the most frequent option for each step in the pipeline, as demonstrated by the choices in bold in Figure 1 and 2. Importantly, the system can be trained on various subcorpora of GLAUx in order to produce different language varieties: the examples above were all trained on Attic prose authors, but if trained on authors such as Herodotus and Hippocrates, it is more likely to produce Ionic forms.

There are some additional complications: firstly, verbs do not only use suffixation but also prefixation. One very frequent prefix is the augment (ε-), which expresses past tense. However, a) in poetic texts, in particular epic poetry, the augment is optional, and b) some verbs have multiple options: e.g. for the verb βούλομαι the augment can be ε- or η- depending on the language variety. Luckily, Morpheus also records what augment is used (if any), so I enhanced the pipeline presented in Figure 1 in such a way that for each past tense verb it is decided whether to apply an augment or not and if so, which one, based on augment frequency information collected for the given verb stem. Several verbs also have prefixes that are part of the lemma: e.g. εἰσέρχομαι ‘go into’ combines ἔρχομαι ‘go’ with the prefix ‘εἰς’. Given that this prefix is formally not part of the stem (e.g. the aorist stem of ἔρχομαι is ἐλθ- regardless whether the εἰς is used or not – formally it also occurs before the augment if one

[1] It should be noted that I expanded the knowledge base of Morpheus beforehand by manually adding frequent words in GLAUx that were initially unrecognized by Morpheus. These additions can be found on my personal GitHub (see ‘Supplementary materials’).

[2] Ancient Greek verbal stems are organized among 7 principal parts, each of which can have a different stem, viz. the present active/middle stem, the aorist active/middle stem, the future active/middle stem, the aorist and future passive stem, the perfect active stem, the perfect middle stem, and the future perfect stem. These parts are automatically determined based on the morphological tag.

is present), prefix information needs to be recorded individually. However, this information is also part of the Morpheus analysis (e.g. for the form εἰσέρχομαι it records the stem ἐρχ- and the prefix εἰς) and can therefore be added to the final output.[3]

Secondly, after combining the various parts of the form, an accent needs to be added. Unfortunately, Ancient Greek accentuation is extremely complicated, and fully explaining how accents are handled by this tool would take up too much space. In what follows I will therefore give a succinct explanation of what is happening, and refer the reader to the code of the tool for more details. Ancient Greek verbs generally have a recessive accent (with some exceptions, that are manually specified), meaning that the accent is on the furthest syllable left 'that Ancient Greek permits', and a rule-based approach is used to determine which syllable this is. As for nouns and adjectives, the accent is lexically determined. For each noun and adjective present in Morpheus, it is specified whether it has a recessive accent, an accent on the stem or an accent on the suffix (some lemmas also have multiple options). During the training process, I therefore also collect the frequencies of these three accent classes for a specific inflectional class (e.g. ος/ου) of a specific stem (e.g. λογ) of a specific lemma (e.g. λόγος). This information is then integrated in the pipeline presented in Figure 2, and based on the accent class and various manually coded rules, it is decided which syllable should be accented with which accent.[4]

Finally, Ancient Greek, as is typical for a natural language, has several irregular word forms (e.g. forms of εἰμί 'to be'). These forms are assigned to an 'irregular' inflectional class as such by Morpheus, and they are not internally analyzed (i.e. fields such as 'ending' would simply be empty). To handle these forms, they are recorded in a separate dictionary (e.g. for the third person singular active present indicative of εἰμί, the form ἐστιν occurs 5460 times). During the generation process, the user can specify a frequency threshold N to retrieve forms from this dictionary. If a specific word form occurs at least N times for the given lemma and morphological features in this dictionary, it is automatically used instead of being generated. This process is also used to generate non-inflected words.

4. Evaluation

4.1 Quantitative metrics

To evaluate the performance of this system, I trained 8 generators on various language varieties[5] present in GLAUx:

- **classical Attic prose** (e.g. Plato, Xenophon)
- **Attic drama** (e.g. Sophocles, Aristophanes)
- **Ionic prose** (mainly Herodotus and Hippocrates)
- **'epic' Ionic poetry** (e.g. Homer, Hesiod as well later imitators such as Apollonius Rhodius)
- **Doric texts** combining both prose and poetry (e.g. Archimedes, Theocritus, Alcman)
- **Aeolic poetry** (mainly Sappho and Alcaeus)
- **'lower' Koine** (a lower variety of the post-classical 'common' language, of which the main representative is the Bible)
- **'higher, Atticistic' Koine** (the Koine variety that is very close to classical Attic prose, used by writers such as Lucian).

I evaluated the results using 10 fold cross-validation: the results presented below sum the errors over all the 10 folds (for Aeolic Greek, due to the very low size of the dataset, I used 100 fold cross-validation instead of 10 fold).

For all nouns, adjectives and verbs presented in the test data, I 'regenerated' the form based on the part-of-speech and the lemma of the word. I then tested whether a) the generator could find an appropriate form and b) whether the generated form was identical to the one in the test data.[6] One should note that in the case of b) it is not always the case that the generator is wrong: since these language varieties are not internally completely homogeneous, it could be the case that both the generated and the attested form is correct, even though they differ – this will be discussed in more detail below. The results are presented in Table 1.

[3] There are some complications, however. Firstly, some phonological rules may modify the appearance of the prefix. The prefix συν- ('with'), for example, becomes συλ- before a stem ending in λ- (e.g. συλλαμβάνω 'to take together'). I manually coded these rules. Additionally, the field 'prefix' in Morpheus always contains the prefix in the Attic standard (the major language variety), even if there is variation, e.g. the Ionic form ἐσέρχομαι would still have εἰς in the prefix field, instead of the Ionic prefix ἐς. In the future I aim to solve this issue by determining what the most frequent appearances of the various prefixes are in the subcorpus that the generator is trained on.

[4] These accent rules are general for all Greek language varieties except Aeolic, which always has a recessive accent on nouns and adjectives as well. This information is not present in Morpheus: all nouns/adjectives only have their default (i.e. non-Aeolic) accent classes recorded, and not an additional recessive accent class for Aeolic. This could be solved by specifying an additional option to the generator to accentuate all words recessively in all cases.

[5] I use the term 'language variety' instead of the more common term 'dialect', since Greek literary dialects do not correspond to how the term is used in present day linguistics: they are literary conventions tied to specific text genres, and often combine forms of various dialects in the conventional sense. See Tribulato (2010) for more details.

[6] I did not evaluate cases of elision (since the generation is context independent, the generated form would always be wrong) and crasis. I also considered forms of εἰμί and φημί that only differ with regard to accent to be identical, as well as forms with and without a movable nu.

Variety	N	Generated	Identical
Attic Prose	1140647	1118510 (98%)	1078372 (95%)
Attic Drama	174110	160240 (92%)	143766 (83%)
Ionic Prose	306447	288668 (94%)	264836 (86%)
Epic Ionic	267688	250934 (94%)	212499 (79%)
Doric	61616	53268 (86%)	46857 (76%)
Aeolic Poetry	3023	989 (33%)	705 (23%)
‘Lower’ Koine	1538676	1494184 (97%)	1451666 (94%)
‘Attic’ Koine	3212551	3152317 (98%)	3042476 (95%)

Table 1: quantitative evaluation of various generators

Unsurprisingly, the most widely attested language varieties were also the ones for which most forms could be generated, viz. classical Attic prose, the ‘lower’ Koine and the ‘Attic’ Koine – all other varieties were trained on less than a million tokens. Whether a form could be generated is dependent on frequency. In some cases the lemma was completely absent from the training data: e.g. for the three varieties discussed above, 37-46% of the forms that cannot be generated occur only once in the subcorpus of the specific variety, i.e. it only occurred in the test fold that was evaluated and never in any of the training folds. In other cases some morphological information (stem, inflectional class or ending) was missing to generate the correct form: for example, one form that could not be generated by the classical Attic prose generator was the future infinitive middle of ἐνθυμέομαι, viz. ἐνθυμήσεσθαι, simply because the verb was never attested in the future stem in the training corpus of this variety.

Even though the training corpus of Doric was quite small (only 62000 tokens) and not homogeneous at all (‘lyric’ Doric is very different from the Doric used in prose), it performed surprisingly well on the quantitative metrics. However, a large part of the dataset includes texts written by Archimedes, whose mathematical prose is extremely repetitive. While 40% of all test tokens are Archimedes data, only 10% of the tokens that were not possible to generate or were not identical to the generated form were from Archimedes, showing the large part this author played in the relatively high accuracy for Doric.

Finally, the only language variety that scored very low on these metrics was Aeolic. This is not surprising, however, given that the training corpus was extremely small (only 10,000 tokens). Another contributing factor is that Morpheus cannot handle the Aeolic dialect well, so that many forms could not be trained on, since there was no morphological information: of the training corpus, 34% of all form types were unrecognized, which is much higher than in general (13%, see Section 3).

4.2 Error analysis

To get a better idea of what went wrong in cases where the generated and the attested form are not identical, I analyzed a random sample of 160 of such ‘mismatches’ (20 per language variety). As stated above, such a mismatch does not automatically mean that the generator is wrong: these can be cases where variation exists within a specific variety. For the present active infinitive of ἁρμόζω for the Attic prose variety, for example, the generator produced ἁρμόττειν while the actual attested form is ἁρμόζειν, both forms which are acceptable in Attic and occur in the Attic subcorpus. Authors may also produce forms that are not ‘proper’ for the specific variety of the text (e.g. when quoting another text, or for example to represent different accents of different characters in a dramatic play – for Attic drama in particular, many of the ‘wrong’ forms were choral lyric where the characters use a pseudo-Doric variety). In fact, 106/160 mismatches (66%) belonged to this category. ‘Epic Ionic’ in particular is a mix of several dialects, so that the ‘non-identical’ forms that were generated were not necessarily incorrect: for 18/20 mismatches of this language variety this was the case.

A related case is the verb ἵστημι (and its compounds), which appeared 7 times in the mismatches. ἵστημι has two aorist stems in Ancient Greek in general, viz. a transitive one (στησ-, meaning ‘to place’) and an intransitive one (στη-, meaning ‘to stand’). Since the morphological tag of GLAUx did not specify transitivity, these were cases where the transitive form was generated but the intransitive attested or vice versa. Again, in such cases the generator was not strictly wrong.

There were an additional 11 cases where the differences in generated and attested form simply related to spelling conventions, viz. the representation of the iota after a long vowel sound (θέληι vs θέλῃ) and the use of diaeresis (γένεϊ vs. γένει). The use of these diacritical marks is simply an editorial decision, so again, in these cases both the generated and attested form are correct.

Finally, in 6 cases there was simply a problem in the GLAUx corpus data, where the form had received the wrong lemma or morphological tag. While the generated form was the one we would expect with this particular lemma and morphological tag, as a consequence it did not match the actual attested form, which should have received a different lemma or morphology.

Moving to the ‘real’ mistakes, in 11 cases the problem was caused by the fact that I did not take into account how degrees of comparison are represented in Morpheus, which often encodes the comparative or superlative form through the stem. For example, for the lemma ἀμβλύς (‘blunt’) and the morphological specification {adjective, singular, neuter, accusative, superlative} the comparative form ἀμβλύτερον was generated instead of the correct superlative form ἀμβλύτατον. This happened because, following the process detailed in Figure 2, the most frequent stem was selected regardless of the degree of comparison which for ἀμβλύς coincidentally happened to be the comparative stem ἀμβλυτερ-. This problem could therefore be solved by making stem selection dependent on the degree of comparison specified in the morphological tag.

The other problems were rather diverse: 7 cases related to problems with the underlying Morpheus database, either because it recorded the wrong stem, or because the inflectional class it specified was not always suitable to generate the correct form[7]; in 7 cases there was a bug in the process that assigned the correct accent to the final generated form or combined the prefix with the rest of the word; in 3 cases the prefix of the verb was not appropriate for the particular dialect of the generator (see footnote 3 above). Finally, there was 1 case related to a particular lemma (τίθημι 'place') where the generative process described in Section 3 was too general: a non-existing form παραθήκαντος was generated for the masculine genitive singular aorist active participle instead of the correct form παραθέντος. This happened because the generator selected the most frequent aorist stem θηκ-, which in fact only occurs in the indicative of παρατίθημι (while in the participle only the stem θ- is possible). Since only tense/voice is taken into account during stem selection, given that in only rare cases mood plays a role in this process in Ancient Greek, a form with a stem inappropriate for the particular mood was generated.

5. Applications

5.1 Generation of morphological paradigms

Many grammars of Ancient Greek offer tables with morphological paradigms as a learning aid. However, a) almost all grammars are tailored to the Attic (or occasionally Koine) language variety and b) these tables are typically not constructed using data from corpora (it is likely that many of them simply copy from each other, under the assumption that Ancient Greek morphology is 'generally known'). Using a generator trained on corpus data, it will be possible to create a more accurate picture of actual usage within a specific language variety. In this context, this tool can be situated in the backdrop of the Pedalion project (Van Hal and Anné, 2017; Keersmaekers et al., 2019) which has adopted such a corpus-based approach in Ancient Greek syntax and vocabulary learning as well.

Some examples of automatically generated tables are provided below (Table 2-4). While these tables provide a single form for the sake of simplicity, through the frequency information learned by the generator it would also be possible to provide multiple forms. For example, following the example in Figure 1, it is possible to generate εἶπον as the dominant aorist of λέγω in Attic prose (to which one could assign a 95% probability, if the relative frequencies of the stem εἰπ- and the inflectional class aor1 are multiplied), but also list εἶπα (with a 2% probability) and ἔλεξα (with a 3% probability) as infrequent alternatives, allowing a much more fine-grained picture of Ancient Greek morphology than the rigid structure of typical grammar textbooks. In other words, both through the generative capabilities of the tool (i.e. it can generate forms even if they are not strictly attested, as discussed in Section 3) and the frequency information it records, a more accurate picture of language usage in Ancient Greek will be provided.

It is important to point out that these tables are based on data from literary dialects, which do not strictly correspond to the spoken (or written epigraphic) dialects at the time (see Tribulato 2010): e.g. 'Doric' in this case means 'forms that prose writers such as Archimedes or a poet who uses the Doric dialect (e.g. Alcman, Stesichorus) are likely to produce'.

	Attic	**Aeolic**	**Doric**	**Koine**
1 sg	ἐθέλω	θέλω	θέλω	θέλω
2 sg	ἐθέλεις	θέλεις	θέλεις	θέλεις
3 sg	ἐθέλει	θέλει	θέλει	θέλει
1 pl	ἐθέλομεν	θέλομεν	θέλομες	θέλομεν
2 pl	ἐθέλετε	?	?	θέλετε
3 pl	ἐθέλουσιν	θέλοισι	θέλοντι	θέλουσιν

Table 2: The present active indicative of the verb (ἐ)θέλω in various language varieties. Since the Doric and Aeolic data did not contain an example of a second plural ending for the inflectional class 'ω stem', this form could not be generated.

	Attic	**Ionic**	**Doric**	**Epic**	**Koine**
Pres.	ὁρῶ	ὁρέω	ὁρᾶ	ὁρόω	ὁρῶ
Aor.	εἶδον	εἶδον	εἶδον	ἴδον	εἶδον
Fut.	ὄψομαι	ὄψομαι	ἰδησῶ	ὄψομαι	ὄψομαι
Perf.	ἑώρακα	ὄπωπα	ὄπωπα	ὄπωπα	ἑώρακα
Mid. pf.	ὦμμαι	ἑώραμαι	?	?	ἑώραμαι
Pass.	ὤφθην	ὤφθην	?	?	ὤφθην

Table 3: Principal parts of the verb ὁράω in various language varieties.

	Attic	**Ionic**	**Doric**	**Epic**
nom sg	τιμή	τιμή	Τιμά	τιμή
gen sg	τιμῆς	τιμῆς	τιμᾶς	τιμῆς
dat sg	τιμῇ	τιμῇ	τιμᾷ	τιμῇ
acc sg	τιμήν	τιμήν	Τιμάν	τιμήν
nom pl	τιμαί	τιμαί	Τιμαί	τιμαί
gen pl	τιμῶν	τιμέων	τιμᾶν	τιμάων
dat pl	τιμαῖς	τιμῇσι	τιμαῖς	τιμῇσι
acc pl	τιμάς	τιμάς	Τιμάς	τιμάς

Table 4: Conjugation of τιμή in various language varieties.

5.2 Masking morphological properties of Ancient Greek words

For an annotation project, our annotators had to label aspectual properties of a given Ancient Greek verb form, i.e. if a verb was an atelic state or activity, or a telic accomplishment or achievement. In order for the annotators not to be influenced too much by the verbal stem (aorist or present) that was chosen (i.e.

[7] For example, for the singular nominative of the lemma Σύλλας the generator trained on the 'Lower' Koine generated Σύλλης instead of the correct Σύλλας. This is because Morpheus assigns both nouns ending in -ης and -ας to the same inflectional class (ης_ου), and the generator therefore simply combined the stem Συλλ- with the most common nominative masculine singular ending of the class ης_ου which was -ης.

they would label aorist verbs as telic and present verbs as atelic regardless of the actual usage context), for each form in one of these two aspects I used a generator trained on the appropriate language variety to create a plausible form in the alternative aspect. An example is as follows (from Lucian Philopseudes, section 33):

τάχα γὰρ ἂν καὶ σύ, ὦ Τυχιάδη, ***ἀκούων / ἀκούσας προσβιβάζοιο / προσβιβασθείης*** *πρὸς τὴν ἀλήθειαν τῆς διηγήσεως.*

While the actual attested forms are ἀκούων and προσβιβασθείης, by generating those two alternatives the annotators can annotate more objectively without knowing that the former form is in the present stem and the latter in the aorist stem.

5.3 Language variety identification

For a final experiment, I tested whether this tool could also be employed in order to identify in which language variety a text (or part of a text) is. As a test case, I focused on Aristophanes, who wrote his comedies primarily in Attic, but many characters speak a non-Attic dialect. To identify these characters, I trained 7 generators on various language varieties (‘Epic’ Ionic, Aeolic poetry, Doric prose, Doric prose and poetry, Ionic prose, Attic drama and Attic prose) and used them to ‘regenerate’ the language of each individual speaker in each individual comedy of Aristophanes: for each token I generated a form through the respective generator based on its lemma and morphological tag. I then checked the Levenshtein edit distance (see Heeringa and Prokić, 2017: 333-334) between the actual attested forms of the speaker and the generated forms, under the assumption that this distance would be lower if the dialect of the speaker ‘matched’ the dialect that the generator was trained on.

Unsurprisingly, of the 195 speakers, for most speakers the edit distance was lowest with forms generated by a generator trained on Attic drama (123 speakers) or Attic prose (29). Of the other generators, the ones trained on Doric (either prose or a combination of poetry and prose) had the second most matches (15), so I manually checked these cases (since I did not check the language of all speakers manually, I can only provide results in terms of precision and not recall), presented in Table 5.

The results were somewhat mixed. 9/15 speakers had at least 100 tokens that could be evaluated, and each of them spoke a non-Attic dialect. 7 of them were Doric (the Megarian of the Acharnians; the poet of the Birds; Lampito, a Laconian, a Laconian herald and the chorus of Laconians in Lysistrata; the chorus of frogs in the Frogs). 2 of them used another dialect: the Scythian archer in the Thesmophoriazusae – since there was no generator trained on ‘Ancient Greek with a strong Scythian accent’, obviously the language variety could never be successfully identified – and a Boeotian in the Acharnians (which is Aeolic, but Boeotian is quite different from the Lesbian Aeolic of Sappho and Alcaeus, and the quality of the Aeolic generator is also not very good, as discussed above). The 6 other speakers all had less than 100 evaluated tokens, and all of them actually spoke Attic instead of Doric. These (very limited) results suggest that this tool has potential to be employed for language variety identification, but only for longer text parts and the results still need to be manually evaluated.

Speaker name	**Text**	**N**	**Dialect**
Megarian	Acharnians	405	Doric
Scythian Archer	Thesmophor.	316	‘Scythian’
Lampito	Lysistrata	171	Doric
Chorus (Lacon.)	Lysistrata	158	Doric
Boeotian	Acharnians	145	Aeolic
Poet	Bird	132	Doric
Laconian	Lysistrata	125	Doric
Chorus (frogs)	Lysistrata	110	Doric
Laconian herald	Lysistrata	105	Doric
Herald	Acharnians	91	Attic
Crestmaker	Peace	35	Attic
Spearmaker	Peace	23	Attic
Old man	Lysistrata	14	Attic
Servant	Clouds	11	Attic
Chaerephon	Clouds	5	Attic

Table 5: Speakers in Aristophanes whose language had the closest Levenshtein distance with forms generated by the 'Doric' generator.

This method may also improve the quality of the language variety identification of the GLAUx texts, since this was often done in a quick and automatic way based on the author of a text (and many texts are still unidentified). Provided that the results are manually checked, this can lead to a possible feedback loop by further improving the quality of the generator (and conversely, then improving the extent to which this tool can be used for language variety identification), given that these labels are used to identify which subcorpus the respective generator should be trained on. For example, initially all texts by Lucian were annotated with the variety ‘high, Atticistic Koine’. However, when using the same method on these texts, there were two texts that showed a better match with the variety ‘Ionic prose’ based on edit distance. These were Lucian’s *On the Syrian Goddess* and *On Astrology*, both of which were actually written in an imitation of classical Ionic prose.

6. Conclusions

This paper has presented a project that aims to automatically generate Ancient Greek word forms in various language varieties, combining the advantages of the rich knowledge base that *Morpheus* has to offer with statistical evidence of a large corpus. While the tool showed decent results, especially for more commonly attested varieties, there are still various limitations that need to be addressed.

Firstly, the quality is highly dependent on the size of the training data. For some dialects, such as Aeolic, this issue cannot be resolved by adding more data, since Aeolic literary production is mainly limited to the texts that were already included in GLAUx. One could possibly add epigraphic data, since there are several inscriptions written in the Aeolic dialect as well. Since there is not a high-quality annotated epigraphic corpus of Ancient Greek at the moment, however, and Morpheus does perform very poorly on inscriptions, this is not a problem that is easily remediable at the moment.

Secondly, one serious bottleneck is the reliance on Morpheus. Several features of Morpheus were hard-coded (e.g. accentuation rules, prefix appearances), which had to be reconstructed for this tool and still likely contain some errors. The tool is also highly dependent on how Morpheus defines a ‘stem’, ‘ending’ and ‘inflectional class’, which might not always match with actual Ancient Greek usage. Additionally, several forms, especially in non-Attic dialects, were still unrecognized by Morpheus. One possible solution is to expand Morpheus’ knowledge base, although this would be a considerable effort. One could also try automated methods: while there are reasons to suspect that solely relying on machine learning may not be feasible for Ancient Greek morphology generation (see Section 2 as well as the introduction of this paper), one could still try to use machine learning to identify the various subparts of a word, as Morpheus does, i.e. the stem, ending and inflectional class. Even if such a prediction is not perfect, it might still lead to better results if such a method can considerably expand the amount of data that the tool can be trained on.

Finally, the tool is currently trained on a subcorpus of texts, of which it is assumed that they constitute a coherent language variety. Apart from misclassifications for the GLAUx texts (which will still undoubtedly be present), this assumption does not always hold (also ignoring the fact that ‘coherent language varieties’ are typically an idealization of the linguistic reality): many GLAUx texts contain multiple varieties, for example pseudo-Doric choral lyric in Attic drama, or quotations of Homer by various authors. There are two possible ways to approach this problem. On the one hand, if this tool is expected to simply generate forms that e.g. a classical Attic prose writer or a Doric poet is expected to produce, the problem disappears to a large extent, given that these authors typically used multiple dialects, as discussed above, and these different possibilities are simply represented in the frequency information the generator has learned from the subcorpus (in other words, the fact that the generator may generate forms from various dialects with various probabilities reflects the actual language situation).[8] On the other hand, if one expects the generator to only produce forms that belong to one clearly demarcated language variety, a more fine-grained method is necessary. For example, one could tackle this problem by training the tool on sentences rather than texts, but it is still an open question which method would be most suitable to identify the language variety of a sentence.

7. Supplementary materials

All code and datasets produced by this research can be found on GitHub (https://github.com/alekkeersmaekers/greek-morphology-generation).

8. Bibliographical References

Beemer, S., Boston, Z., Bukoski, A., Chen, D. Dickens, P., Gerlach, A., Hopkins, T., Jawale, P. A., Koski, C., Malhotra, A., Mishra, P., Muradoğlu, S., Sang, L., Short, T., Shreevastava, S., Spaulding, E., Umada, T., Xiang, B., Yang, C., and Hulden, M. (2020). Linguist vs. Machine: Rapid Development of Finite-State Morphological Grammars. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 162–170, Online, July. Association for Computational Linguistics.

Canby, M. and Hockenmaier J. (2023). A Framework for Bidirectional Decoding: Case Study in Morphological Inflection. In *Findings of the Association for Computational Linguistics (EMNLP 2023)*, pages 4485–4507, Singapore, December. Association for Computational Linguistics.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Walther, G., Vylomova, E. McCarthy, A. D., Kann, K., Mielke, S. J., Nicolai, G., Silfverberg, M., Yarowsky, D., Eisner, J., and Hulden, M. (2018). The CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection. In *Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection*, pages 1–27, Brussels, October. Association for Computational Linguistics.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Walther, G., Vylomova, E., Xia, P, Faruqui, M., Kübler, S., Yarowsky, D., Eisner, J., and Hulden, M. (2017). CoNLL-SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection in 52 Languages. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 1–30. Vancouver, August. Association for Computational Linguistics.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Yarowsky, D., Eisner, J., and Hulden, M. (2016). The SIGMORPHON 2016 Shared Task—Morphological Reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 10–22, Berlin, August. Association for Computational Linguistics.

Crane, G. (1991). Generating and Parsing Classical Greek. *Literary and Linguistic Computing* 6(4):243–45.

[8] One could criticize this by stating that it is not merely a matter of frequency: the actual language context also plays a role (i.e. we expect only Doric in the choral parts of Attic drama). However, in this paper I only explored context-independent morphological generation: if the actual linguistic context needs to be taken into account, other problems such as phonological appropriateness also need to be tackled.

Goldman, O., Batsuren, K., Khalifa, S., Arora, A., Nicolai, G., Tsarfaty, R., and Vylomova, E. (2023). SIGMORPHON–UniMorph 2023 Shared Task 0: Typologically Diverse Morphological Inflection. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 117–125, Toronto, July. Association for Computational Linguistics.

Hajic, J. (2000). Morphological Tagging: Data vs. Dictionaries. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 94–101, Seattle, May. Association for Computational Linguistics.

Heeringa, W. and Prokić, J. (2017). Computational Dialectology. In C. Boberg, J. Nerbonne, & D. Watt (Eds.), *The Handbook of Dialectology*. Hoboken, NJ: Wiley Blackwell, pp. 330–347.

Kanerva, J, Ginter, F., and Salakoski, T. (2021). Universal Lemmatizer: A Sequence-to-Sequence Model for Lemmatizing Universal Dependencies Treebanks. *Natural Language Engineering* 27(5):545–74.

Keersmaekers, A. (2020).“Creating a Richly Annotated Corpus of Papyrological Greek: The Possibilities of Natural Language Processing Approaches to a Highly Inflected Historical Language. *Digital Scholarship in the Humanities* 35(1):67–82.

Keersmaekers, A. (2021). The GLAUx Corpus: Methodological Issues in Designing a Long-Term, Diverse, Multi-Layered Corpus of Ancient Greek. In *Proceedings of the 2nd International Workshop on Computational Approaches to Historical Language Change 2021*, pages 39–50, Online, August. Association for Computational Linguistics.

Keersmaekers, A, Mercelis, W., Swaelens, C., and Van Hal, T. (2019). Creating, Enriching and Valorizing Treebanks of Ancient Greek. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 109–17, Paris, August. Association for Computational Linguistics.

Kirov, C., Sylak-Glassman, J., Que, R., and Yarowsky, D. (2016). Very-large Scale Parsing and Normalization of Wiktionary Morphological Paradigms. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3121–3126, Portorož, May. European Language Resources Association (ELRA).

Kodner, J., Khalifa, S., Batsuren, K., Dolatian, H., Cotterell, R., Akkus, F., Anastasopoulos, A., Andrushko, T., Arora, A., Atanelov, N., Bella, G., Budianskaya, E., Ghanggo Ate, Y., Goldman, O., Guriel, D., Guriel, S., Guriel-Agiashvili, S., Kierás, W., Krizhanovsky, A., Krizhanovsky, N., Marchenko, I., Markowska, M., Mashkovtseva, P., Nepomniashchaya, M., Rodionova, D., Sheifer, K., Serova, A., Yemelina, A., Young, J., and Vylomova, E. (2022). SIGMORPHON–UniMorph 2022 Shared Task 0: Generalization and Typologically Diverse Morphological Inflection. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 176–203, Seattle, July. Association for Computational Linguistics.

McCarthy, A. D., Vylomova, E., Wu, S., Malaviya, C., Wolf-Sonkin, L., Nicolai, G., Kirov, C., Silfverberg, M., Mielke, S. J., Heinz, J., Cotterell, R., and Hulden, M. (2019). The SIGMORPHON 2019 Shared Task: Morphological Analysis in Context and Cross-Lingual Transfer for Inflection. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–244, Florence, August. Association for Computational Linguistics.

Merzhevich, T., Gbadegoye, N., Girrbach, L., Li, J., and Soh-Eun Shim, R. (2022). SIGMORPHON 2022 Task 0 Submission Description: Modelling Morphological Inflection with Data-Driven and Rule-Based Approaches. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 204–211, Seattle, July. Association for Computational Linguistics.

Pimentel, T., Ryskina, M., Mielke, S. J., Wu, S., Chodroff, E., Leonard, B., Nicolai, G., Ghanggo Ate, Y., Khalifa, S., Habash, N., El-Khaissi, C., Goldman, O., Gasser, M., Lane, W., Coler, M., Oncevay, A., Montoya Samame, J. R., Silva Villegas, G. C., Ek, A., Bernardy, J.-P., Shcherbakov, A., Bayyr-ool, A., Sheifer, K., Ganieva, S., Plugaryov, M., Klyachko, E., Salehi, A., Krizhanovsky, A., Krizhanovsky, N., Vania, C., Ivanova, S., Salchak, A., Straughn, C., Liu, Z., Washington, J. N., Ataman, D., Kieraś, W., Woliński, M., Suhardijanto, T., Stoehr, N., Nuriah, Z., Ratan, S., Tyers, F. M., Ponti, E. M., Aiton, G., Hatcher, R. J., Prud'hommeaux, E., Kumar, R., Hulden, M., Barta, B., Lakatos, D., Szolnok, G., Ács, J., Raj, M., Yarowsky, D., Cotterell, R., Ambridge, B., and Vylomova, E. (2021): SIGMORPHON 2021 Shared Task on Morphological Reinflection: Generalization Across Languages. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–259, Online, August. Association for Computational Linguistics.

Tauber, J. (2016). greek-inflexion. Python. https://github.com/jtauber/greek-inflexion.

Tribulato, O. (2010). Literary Dialects. In Bakker, E. J. (Ed.), *A Companion to the Ancient Greek Language*. Chichester: Wiley, pp. 388–400.

Van Hal, T., and Anné, Y. (2017). Reconciling the Dynamics of Language with a Grammar Handbook: The Ongoing Pedalion Grammar

Project. *Digital Scholarship in the Humanities* 32(2):448–54.

Vatri, A., and McGillivray, B. (2020). Lemmatization for Ancient Greek: An Experimental Assessment of the State of the Art. *Journal of Greek Linguistics* 20(2):179–96.

Vylomova, E., White, J., Salesky, E., Mielke, S. J., Wu, S., Ponti, E. M., Maudslay, R. H., Zmigrod, R., Valvoda, J., Toldova, S., Tyers, F., Klyachko, E., Yegorov, I., Krizhanovsky, N., Czarnowska, P., Nikkarinen, I., Krizhanovsky, A., Pimentel, T., Torroba Hennigen, L., Kirov, C., Nicolai, G., Williams, A., Anastasopoulos, A., Cruz, H., Chodroff, E., Cotterell, R., Silfverberg, M., and Hulden, M. (2020). SIGMORPHON 2020 Shared Task 0: Typologically Diverse Morphological Inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 1–39, Online, July. Association for Computational Linguistics.

Decoding Byzantine Book Epigrams: an Exploration of Machine-assisted Extraction of Formulaic Material

Kyriaki Giannikou[†], Colin Swaelens[‡], Ilse De Vos[*], Els Lefever[‡], Klaas Bentein[†]
[†]Dpt. of Linguistics, [‡]Language & Translation Technology Team, [*]Flanders AI Academy
[†]Blandijnberg 2, Ghent, Belgium [‡]Groot-Brittannielaan 45, Ghent, Belgium
[*]Kasteelpark Arenberg 10, Leuven, Belgium
{author1, author2, author4, author5}@ugent.be
author3@kuleuven.be

Abstract

This paper proposes a machine-assisted methodology for identifying and extracting formulaic sequences from a subset of the Database of Byzantine Book Epigrams (DBBE). The methodology involves conceptualising formulaicity within the DBBE corpus, pre-processing and extracting n-grams from textual data, followed by refinement before delving into the interpretation of the results. Through systematic application of this methodology, some initial insights into the characteristics of formulaic language within the Byzantine book epigram tradition are gained. Representative findings illustrate the nature of recurring patterns, cases of creative elaboration, and their content. This initial exploration aims to facilitate a deeper understanding of the concept of formulaicity in Byzantine book epigrams; while computational analysis provides a quantitative perspective, linguistic and philological research is necessary for a more nuanced understanding. Future research directions include refining the methodology and expanding the scope of analysis beyond the current subset of the DBBE. Overall, this study lays the groundwork for further research on this rich book epigram tradition.

Keywords: Byzantine Greek, Corpus Linguistics, Natural Language Processing

1. Introduction

Formulaic language plays a pivotal role in linguistic communication across various domains, from TV shows to religious rituals, or regal ceremonies; formulaicity is omnipresent. It is characterised by recurrent sequences of varying degrees of fixedness and compositionality, ranging from collocations to idioms. Its study traverses various linguistic disciplines, while it has also captured the interest of scholars in literary studies.

This paper delves into the rich tapestry of formulaic language within the context of the Database of Byzantine Book Epigrams (DBBE) corpus, a repository teeming with paratextual material that offers invaluable sociolinguistic insights into Byzantine society's book culture (Ricceri et al., 2023). A Byzantine book epigram is defined as a poem, a text in verse, found in the margins of Byzantine manuscripts; it is written in a book and refers to the book's production and consumption (Kominis, 1966; Lauxtermann et al., 2003; Bernard and Demoen, 2019). In general, book epigrams are written by scribes who are copying the main text of a manuscript and provide the reader with more insights into that particular manuscript. In Example 1, found at the end of a manuscript, the scribe breathes a sigh of relief that the copying of the text is finished. DBBE Occurrence 17571 is just one attestation, however, of a whole series comprising over 200 epigrams. These vary in length, ranging from one to four verses, and are characterised by the employment of different words and different word orders.

Given that some book epigrams are attested over a hundred times, being copied, adapted, or extended by other scribes, it is argued here that book epigrams establish a fully-fledged literary genre. This copying results in repetition of half, one or multiple verses, or even complete epigrams, which brings us to the hypothesis that this corpus of book epigrams displays noticeable formulaicity. Although this claim is an admitted truth (Bernard and Demoen, 2019; Ricceri et al., 2023), a comprehensive linguistic study of the formulaic and creative aspects of book epigrams has only recently begun. This paper marks the initial step of this endeavour: the machine-assisted extraction and identification of formulaic material from this particular corpus.

(1) Ὥσπερ ξένοι χαίρουσιν ἰδεῖν πατρίδα,
οὕτως καὶ οἱ γράφοντες βιβλίου τέλος.
Hōsper xenoi chairousin idein patrida
houtōs kai hoi graphontes bibliou telos.
Just like travellers rejoice upon seeing their homeland, so do writers at the end of a book.[1]
DBBE Occurrence 17571

Our exploration is anchored in a multidisciplinary approach, drawing from computational methodologies, historical linguistics, and literary conventions to unearth the formulaic material embedded within these epigrams.

[1] All translations are provided by the authors.

Our undertaking begins with an examination of relevant literature (Section 2). A detailed description of the DBBE data and our selected subset follows in Section 3. The method to extract and identify formulaic sequences is given in Section 4. Our methodology comprises a series of systematic steps, from data export and pre-processing to n-gram extraction and dictionary creation, leading to a manual evaluation of the results of the proposed machine-assisted method. This is followed by a discussion of the results (Section 5) and a conclusion with a glimpse in future research (Section 6) on capturing the formulaic language employed by scribes.

2. Literature Review

2.1. Formulaicity

Linguistic research on the phenomenon of formulaicity spans across various disciplines, including corpus linguistics, psycholinguistics, cognitive linguistics, and sociolinguistics. The initial systematic description of this phenomenon can be traced back to the 19th century and the observations of standardised speech production among aphasic patients. Subsequent contemporary studies have placed significant emphasis on the social dimension of formulaic language. While the exact terminology may vary among theorists, linguists generally agree on a continuum of formulaicity, with formulaic sequences ranging from collocations to idioms. This continuum is based on the degree of lexical, morphological and syntactic fixedness (i.e. the extent of variation permitted) and compositionality (i.e. whether the formula can be deconstructed into smaller, meaningful components).

Formulaicity is widely recognised as an inherent characteristic of language. Sinclair (1991) highlighted fixed linguistic sequences or *collocations* as prevalent in English corpora. As outlined in his *idiom* and *open-choice principles*, he attributed this prevalence to cognitive efficiency. There he also proposed a computational method for extracting collocations from large corpora, a methodology that has been successfully applied to various contemporary languages, e.g. the Bantu languages (de Schryver, 2008).

Wray (2002, 2008) has made significant contributions to the field. Her Morpheme Equivalent Units (MEUs) model suggested that prefabricated sequences are both stored and retrieved as a whole from the mental lexicon. This occurs unless there is a need for parsing, according to the Needs-Only Analysis (NOA) model she proposed. These prefabricated 'chunks' can be either *fully-fixed* or semi-preconstructed (*partially-fixed frames*). *Partially-fixed frames* can either allow more specific slots (i.e. very detailed variation) to be completed or less specific ones. An example of the former is variation in tense or mood of a verb, the latter is a placeholder that can hold any word at that specific slot. Formulaicity serves both social and cognitive functions in communication, according to Wray; it reduces processing effort for participants, aids in the manipulation of the hearer and contributes in marking discourse structure. Wray (2008) also discussed commonly used criteria for identifying formulaic sequences. These criteria include their frequency of occurrence, perceived formulaicity based on native speakers' intuition, phonological or spelling indicators of holistic processing or chunking, and fixedness – all of which, in her opinion, present weaknesses if used in isolation rather than in combination. Additionally, she proposed eleven diagnostic criteria for evaluating intuitive judgements.

Kuiper (2000) emphasised external, cultural factors, such as the need for socially accepted and predictable communication, along with the internal factor of memory constraints, as motivation for formulaic language use. Later, Kuiper (2009) defined formulae as 'lexical items with the features $[-\text{wlc} + \text{nlcu}]$'. This means that they cannot be broken down into meaningful sub-units, resulting in no Word-Level Complexity (wlc) – at least not retaining all their possible compositional meanings – and they do have specific Non-Linguistic Conditions of Use (nlcu), including 'speech community' and occasion. Most importantly, he introduced the concept of *formulaic genres* (e.g. weather forecasts), recognising that formulaic elements characterise larger discourse structures as formulaic in their own way, emphasising the interrelation of formulaic language with broader communicative genres and underscoring its role in discourse marking.

From a constructionist perspective, formulaic sequences are conceptualised as *form-function pairings*, i.e. constructions (Goldberg, 1995, 2006; Buerki, 2016b). The analysis of formulae as constructions, particularly of those that are more abstract, led Buerki (2016b) to regard the distinction between formulae and other constructions arbitrary. However, he still asserted that the psycholinguistic significance of formulaic sequences, indicated by Wray (2002), persists.

Research on formulaicity of course extends beyond English and includes a multitude of other modern languages; just to give some examples, formulaic sequences have been identified and extracted from Swedish (Cinková et al., 2006), Slovenian (Dobrovoljc, 2020), and Spanish (Cortes, 2022) corpora, using (semi-)automatic procedures. Buerki (2016a) provided a general guide to such procedures using software tools like N-Gram Processor[2]

[2] https://github.com/buerki/ngramprocessor

and Sub-String[3].

2.2. Historical Corpora

Recent linguistic research on formulaicity has broadened its scope to include historical corpora. For instance, Moulin et al. (2015) outlined the study of formulae in historical German (beginning of written tradition to the Early New High German period), while Rutten and van der Wal (2012) applied Wray's insights to a historical corpus of Dutch letters dating from the 17th and 18th centuries. The latter argued for the pertinence of formulaicity research, notwithstanding the non-contemporary and purely written nature of the corpus. Their findings indicated that formulaic language serves to compensate for limited literacy and writing experience within written contexts, just like it reduces processing effort within an oral context. This aligns with earlier observations that compared to more experienced speakers and performers, less experienced ones tend to rely more heavily on formulae, thereby enhancing their fluency (Lord, 1960; Leiwo, 2005; Bozzone, 2010). It should be noted that the particularities of their historical corpus necessitated adaptation; the recurrent references to the Christian god led to the establishment of a distinct *Christian-ritual formula*, complementing Wray's general communication-based functions of formulaic language. This showcases the feasibility of applying such a theoretical framework on a historical, written corpus, notwithstanding the peculiarities imposed by its socio-cultural context.

In the context of the Greek language, scholars have predominantly explored formulaicity and creative variation within literary corpora such as the Homeric poems. Parry and Lord's seminal work on the oral-formulaic theory (Lord, 1960) revealed the mnemonic function of recurring fixed expressions in traditional oral poetry. More recent scholarship (Bakker, 2005; Foley, 2007) has delved into the creative utilisation of formulae within the narratives of the Homeric epics. They have explored how these formulae are not static elements but rather dynamically employed, adapting to the oral tradition and the metrical constraints of the hexameter. Through their analyses, they have highlighted the fluidity and adaptability of formulae, shedding light on their pivotal role in shaping the rich tapestry of Homeric storytelling. Bozzone (2010, 2014), in particular, contributed to the discourse by analysing Homeric formulae as constructions, taking into consideration factors inhibiting formulaic reliance and poetic novelty. Jeffreys (1973), on the other hand, extended the Homeric formula theory to late Byzantine popular poetry, while later works (Jeffreys and Jeffreys, 1983, 1986) highlighted the impact of textual transmission on variation in formulae. Limited studies exist however on formulaic language in earlier Byzantine literary composition, and mostly discussing specific formulaic expressions, often not exhaustively (Garitte, 1962; Treu, 1977; Boeten et al., 2021), necessitating further investigation.

In recent years, there has been a notable shift in scholarly focus from prestigious literary texts to everyday documents, reflecting a broader interest in aspects of daily life. Non-literary (documentary) corpora, both ancient and medieval[4] Greek, are thus receiving more and more attention. The standardised, conventional language they present has led to systematic studies on their formulaic material, particularly within administrative texts, such as contracts, petitions, public inscriptions, and letters, originating from both private and business correspondence (Lazzarini, 1976; Nachtergaele, 2015; Bentein, 2023). The shift to everyday documents also sparked increased interest in paratextual material. Regarding the Byzantine period in particular, manuscripts have evolved beyond their initial perception as repositories of classical texts to being acknowledged as valuable reservoirs of paratextual material, a term coined by Genette (1987). Paratextual material includes elements accompanying the main text to provide information on, e.g., authorship, genre, and purpose. Scholars (Ciotti and Lin, 2016; Teeuwen and van Renswoude, 2018) now focus on paratexts to unveil historical sociolinguistic dimensions of manuscripts. They regard paratexts as vital markers that reveal the sociohistorical context of the manuscript and its intricate interplay with language, intellectuality, culture, and society.

The Database of Byzantine Book Epigrams project (Ricceri et al., 2023) has built, digitised, and made available such a paratextual corpus to the scholarly community, including scholars, linguists, and recently also Natural Language Processing (NLP) researchers and engineers. Byzantine book epigrams, written in the margins of manuscripts, intertwine poetic expression with practical details. As such, they shed light on facets such as the patrons of the manuscripts and the identities of the scribes involved in their transcription. They do so by reproducing or building upon standard, formulaic material, as it has been observed (Bernard and Demoen, 2019).

3. Data

The aim of the DBBE has been the digitisation and enrichment of a distinct textual corpus, specifically focusing on Byzantine book epigrams. Byzantine

[3] https://github.com/buerki/SubString

[4] Byzantine and medieval will be used within this paper as synonyms to address the period of the 5th until the 15th century.

book epigrams are paratextual in nature, as they exist on the threshold between the intellectual realm of the text and the physical manifestation of the book they are inscribed on. Overall, the corpus' significance is derived from its direct association with the material context of manuscripts, providing insights into codicology, palaeography, production, and reading practices in the context of Byzantine society. This makes them a valuable source for understanding the social dynamics of book culture. The DBBE corpus provides an excellent opportunity to shed light on the actors and communities involved in manuscript production and consumption and, therefore, on their formulaic and creative language use. Often originating from non-professional poets, these epigrams offer a glimpse into less erudite literary devices and linguistic developments.

3.1. The DBBE: Occurrences and Types

The organisation of book epigrams in the DBBE involves a categorisation of entries into 'Occurrences' and 'Types'. 'Occurrences' represent instances of epigrams as they appear in manuscripts, accommodating all variation in the original texts as transcribed by the DBBE team, based on the original manuscript and catalogues or related publications. 'Types', on the other hand, constitute reconstructed texts grouping similar 'Occurrences' while accommodating mostly minimal variation. Thus, 'Type' records present normalised, readable versions of the manuscript evidence. Although 'Types' provide scholars with a more homogeneous and standardised corpus in terms of language, this standardisation deprives us of access to the scribes' original linguistic choices and oversimplifies the dynamic nature of textual transmission. Book epigrams undergo copying and reworking, thus being transmitted in similar yet different forms, a phenomenon not clearly illustrated by the standardisation of the 'Types'. However, individual differences are preserved at the level of 'Occurrences' in any case. Table 1 shows that the DBBE holds 12,497 'Occurrences' that are all linked to one or more of the 5,022 'Types'.

	Epigrams	**Verses**	**Tokens**
Occurrences	12,497	48,458	272,426
Types	5,022	24,879	140,103
Scribe Types	1,849	13,150	32,888

Table 1: The number of epigrams, verses and tokens of both the DBBE Types and Occurrences.

Example 2 shows 'Type' 1974 2a and one of its corresponding 'Occurrences' in 2b. This 'Occurrence' is highly affected by itacism, the shift of the classical Athenian pronunciation of four vowels (ι *i*, η *è*, ε *e*, υ *u*) and two diphthongs (ει *ei*, οι *oi*) to one and the same [i] sound. The author of Example 2b clearly knew what he wanted to write, but did not stick to archaising orthographic conventions. That is why this 'Occurrence' is placed under the umbrella of Type 1974.

(2) a. Ἡ μὲν χεὶρ ἡ γράψασα σήπεται τάφῳ
Hè men kheir hè grapsasa sèpetai taphōi
Type 1974 v.1

b. Ἱ μὲν χὺρ ἡ γράψασα σύπτεται (!) τάφω
Hi men khyr hè grapsasa syptetai (!) taphō
Occurrence 18305 v.1

The hand that has written, is rotting in the grave

The rationale behind using DBBE 'Types' instead of 'Occurrences' for the presented research can be summarised as follows. First, at this preliminary stage, the primary objective is to develop a method for extracting recurring material to identify and define what constitutes a formula in the Byzantine Book Epigram corpus. Further exploration into mapping, analysing, and explaining the variation present in formulae will occur in subsequent research phases. Moreover, in the absence of a reliable lemmatisation tool, a degree of standardisation is essential for the initial automatic extraction of collocations, due to the significant morphological and orthographic variation present in Byzantine Greek corpora (cf. Example 2). Furthermore, when considering formulaic sequences at the level of 'Occurrences,' they often vary depending on their contextual information. This contextual information dictates the specific content of 'slots' that need to be filled between the standard, recurring, formulaic elements; e.g. when signing at the end of the work: χεὶρ (ADJ) NAME, 'the hand of (ADJ) NAME' (Example 3). The specific content of these slots is not relevant at this stage of formulaicity research, thus an examination at the level of 'Types' is more appropriate. Lastly, by extracting formulaic material from the 'Types,' we ensure that their frequency of occurrence reflects their usage in various contexts. Thus, the extracted sequences are more likely to constitute formulaic entities themselves, rather than formulations that, for example, were repeatedly copied as part of the exact same epigram. [5]

[5] It is important to note that, while this is mostly true, it is not guaranteed that if a formula appears in two distinct 'Types', the variation between these 'Types' will always be pertinent to the usage of that sequence. As this paper marks the initial exploration into the corpus's formulaicity, it is under the provisional hypothesis that such a correlation exists that we shall proceed. However, it is essential to consider this caveat for future linguistic and philological research.

Epigram entries in the DBBE include relevant metadata, such as the manuscript they are to be found in, the dating of that manuscript, their meter, etc. Of relevance to our discussion is the 'genre' that is attributed to them. The DBBE categorises the corpus into six genres 'based on the main actors that play a role in the communicative situation typical for book epigrams' (Ricceri et al., 2023). In this way, epigrams are divided into 'Text-', 'Author-', 'Scribe-', 'Reader-', 'Image-', and 'Patron-related', with epigrams often belonging to more than one genre. For the purpose of the present endeavour, this paper will focus on those 'Types' tagged as 'Scribe-related'.

The rationale behind this second choice can be summarised as follows. First, based on observations of former and current DBBE scholars, 'Scribe-related' epigrams, which are often referred to as 'metrical colophons', constitute the most standardised, 'formulaic' genre within the DBBE corpus (Bernard and Demoen, 2019). Furthermore, poetic colophons, constituting statements at the end of the book in verse, document information such as the scribe's name, the date of completion, and/or the place of writing. There is a practical need of providing a minimal set of information, which still provides room to poetic licence. Pre-constructed formulations are more likely to fulfil this specific yet standard need, with contextual information filling in 'slots' between standard expressions (see Example3.1). Lastly, metrical colophons constitute a well-studied genre due to their presence in various manuscript traditions, for example in Armenian manuscripts (e.g. Sanjian 1969; van Elverdinghe 2023). Therefore, conducting research on their Byzantine Greek counterparts will provide a solid foundation for future comparisons between these manuscript cultures.

4. Method

To achieve the objective of identifying and extracting formulaic sequences from our dataset, a multi-step methodology was devised. First, a conceptualisation of the DBBE formulae was undertaken, precisely defining the material sought after. Next, textual data was exported from the Scribe-related 'Types' subset of the database. This data then underwent pre-processing, before proceeding with n-gram extraction, wherein contiguous sequences of tokens were identified. These sequences were subsequently compiled into a dictionary, constituting a repository of recurring patterns. In the next stage, the dataset was refined, prioritising formulaic sequences and eliminating redundancy. Finally, manual evaluation was conducted. This entailed systematic comparison and documentation of formulaic material. This comprehensive methodology ensures a systematic and comprehensive approach to identifying and extracting formulaic sequences from the DBBE corpus. It addresses the complexities of the Byzantine book epigram corpus and navigates potential obstacles encountered during the analysis.

4.1. Conceptualisation: formulae in the DBBE

Before proceeding to the machine-assisted extraction of formulaic material from the DBBE corpus, it was crucial to define what constitutes a formula. This definition guided our search for relevant material and ensured consistency in our identification process. As acknowledged in scholarly discourse however, identifying formulae presents a challenge. This is due to the circularity of the task. As Wray (2008) puts it, 'you cannot reliably identify something unless you can define it. (...) In order to establish a definition, you have to have a reliable set of representative examples, and these must therefore have been identified first' (93).

Given that our corpus consists of written poems, distinct from the speech evidence typically used in modern linguistic research and the orality factor in Homeric studies, it was necessary to consider the following implications. In contemporary linguistic corpora, one of the functions of formulaic language is to reduce processing effort for speakers and listeners. This does not apply here as the DBBE corpus comprises written compositions. Instead, as suggested by Rutten and van der Wal (2012), the use of formulaic language in our corpus, and especially the Scribe-related subset, might serve as compensation for limited literacy and writing experience among scribes. Similarly, conclusions about 'speech community' practices or formulaic *speech* production cannot be drawn. Instead, it is possible to draw conclusions on the scribal 'text community' (Stenroos, 2018) and its repertoire, accepting the limitations of our historical corpus.

Considering these factors, it became essential to assess the applicability of common criteria used in linguistic research for identifying formulaic material:

1. Frequency: the frequency of a sequence remains a relevant criterion for the DBBE corpus.

2. Fixedness: while the degree of fixedness is applicable to our material, it is important to note that formulaic material may not always be fully-fixed (see Example 3.1). *Fully-fixed* formulae represent only one aspect of the spectrum of fixedness.

3. Intuition: intuitive judgements of native speakers regarding the formulaicity of sequences do not apply to historical corpora like ours. Although we could apply a notion of intuition,

the resulting conclusions would lack sufficient objectivity. Similarly, assessing compositionality or idiomaticity is challenging for non-native speakers, and depends on their degree of familiarity with the corpus.

4. While phonological aspects are not applicable, spacing and punctuation[6] can provide insights into 'chunking' patterns. However, this requires separate palaeographical analysis.

Based on these considerations, our working definition of formulaic sequences in the DBBE corpus is as follows: **recurring phrases that are integral to the repertoire of the scribal 'text community'**.

4.2. Data Export

First, textual data from the DBBE was extracted. At this preliminary stage, it was determined that exporting all DBBE 'Types' would suffice. As we already mentioned in Section 3, the 'Occurrences' are out of scope for this paper. From the 5,022 'Types', we compiled a collection of the 'Scribe-related Types'. For each of the remaining 1,849 epigrams we created a .txt file containing its text.

4.3. Pre-processing

Although the DBBE 'Types' are standardised, they present Byzantine Greek poems that employ an elaborate accentuation system, comprised of breathings indicated with a *spiritus asper* (rough breathing) or *lenis* (smooth breathing), and accents (acute, circumflex, grave), alongside nuanced punctuation conventions. While these linguistic features are relevant, they do not inherently influence the presence of formulaic material. The same applies to capital letters. However, they may pose challenges in character identification for computational analysis. For instance, distinguishing between Θεός and ϑεὸς, or Θεοῦ and Θ(εο)ῦ (expanded abbreviation), may be irrelevant for our analytical objectives. Therefore, pre-processing of the textual data was performed by removing accentuation, punctuation, and any non-essential formatting to ensure uniformity and enhance computational analysis capabilities.

(3) Χριστέ, ὁ ϑεὸς ἡμῶν χαροποιήσας
Christe, ho theos hèmōn charopoièsas
Christ, our god that causes joy.
DBBE Type 2380

(4) Ἀρχὴ καὶ τέλος ὁ Θεὸς ἡμῶν δόξα
Archè kai telos ho theos hèmōn doxa
The beginning and end, our God, our splendour
DBBE Type 4131

Without this pre-processing, the 3-gram ὁ ϑεὸς ἡμῶν from Example 3 would never match the 3-gram ὁ Θεὸς ἡμῶν from Example 4, even though they differ only in whether or not the thèta is capitalised. When counting n-grams, it is not desirable to miss out on correct matches due to these *irrelevant* linguistic features.

4.4. N-gram extraction

Subsequently, the pre-processed text underwent n-gram extraction. N-grams represent contiguous sequences of n tokens from a given sample of text, such as the 2-gram ὁ ϑεὸς in Example 3 or the 3-gram Ἀρχὴ καὶ τέλος in Example 4. We computed 2- to 12-grams for all 'Types' present in our dataset, providing frequency counts. The rationale behind the maximum n value of the n-grams was to capture the maximum number of words per verse present in our subset, which is 12. Based on our familiarity with the corpus, recurring patterns tend to occupy half a verse, a whole verse, or multiple complete verses. Thus, this approach allowed the identification of formulaic sequences present within verse limits.

At this stage, given the flexible syntax of the Greek language (van Emde Boas et al., 2019), two approaches to n-gram extraction were explored: one that takes into account the word order per verse and another that only considers the presence of words regardless of their order. In the former approach, variation in word order is captured and deemed significant, with the aim of capturing *fully-fixed* formulaic sequences. In contrast, the latter approach focuses on capturing a broader range of formulaic patterns (i.e. beyond the fully-fixed ones), thus assuming that word order is not a significant limiting factor in Byzantine Greek formulaic language as found in book epigrams.

Our familiarity with the corpus supports the latter perspective. The result of this step of the methodology consisted of lists of non-word-order-sensitive n-grams and their corresponding frequency.

4.5. Dictionary creation

Following the extraction of all n-grams, a dictionary comprising them was compiled to facilitate subsequent analysis. This dictionary functioned as a comprehensive repository of recurring sequences within the corpus, sorted from most to least frequently occurring. All combinations of 2 to 12 words

[6] Note that most texts are written in *scriptio continua* (i.e. without separating words using spaces) and the use of punctuation differs from our modern conventions.

that occur more than once in our corpus were included in the dictionary as potentially formulaic. It is noted that sequences with higher frequency counts are more likely to be formulaic. However, it is evident that a sequence of function words (i.e. articles, conjunctions, prepositions, pronouns), [7] cannot – for our purposes at least – feasibly be deemed formulaic, despite their frequent appearance in the corpus. In this paper, these are called *function-word sequences* and considered non-formulaic.

4.6. Last dataset refinement

In order to further refine the dataset and prioritise recurring sequences that are most likely to be formulaic, an additional automatic cleaning process was implemented. This process involved subtracting the frequency of (n+1)-grams that include an n-gram from the frequency of that n-gram. In essence, if an n-gram occurrs within an (n+1)-gram, the frequency count of the (n+1)-gram was deducted from the frequency count of the n-gram. For example, the 3-gram ωσπερ, ξενοι, χαιρουσιν occurs 13 times, while the 4-gram ωσπερ, ξενοι, χαιρουσιν, ιδειν occurs 12 times. Thus, the non-redundant frequency of this 3-gram is 1, as in all other 12 instances it occurs merely as a part of the 4-gram. This adjustment aims to reduce redundancy and enhance the distinction between phrases that are parts of formulaic sequences and complete formulaic units themselves.

4.7. Manual formulaicity evaluation

The final stage of analysis involved a comparative examination of the results based on the new n-gram values.

Interpreting the formulaicity results acquired from the previous step, required the following considerations:

1. *Function-word sequences* are not considered formulaic.

2. 2- and 3-grams containing at least one or two non-function words, respectively, are not considered formulaic; e.g., the bi-gram τε, και (both, and) occurring 119 times, την, βιβλον (the book) occurring 96 times, or η, βιβλος, αυτη (this book here) occurring 49 times. Exceptions are cases that represent a recognisable entity belonging to the Christian repertoire [8]. These can be prepositional phrases, like σὺν Θεῷ *syn Theō* (with (the help of) God), short supplications, like δίδου μοι *didou moi* (give me), or typical vocatives and exclamations, such as Χριστὲ μου *Christe mou* (my Christ) or δόξα Σοι *doxa Soi* (praise to You).

3. Special status is attributed to (>2)-grams containing articles or pronouns that do not modify terms within the (>2)-gram, prepositions without their modifier, or transitive verbs that render the (>2)-gram semantically incomplete without an object. These are considered potential *open-slot formulae*. The term *open-slot formula* will be henceforth used to refer to formulaic material that includes placeholders (X) to be filled based on the occasion (e.g. δόξα τῷ Θεῷ τῷ X (adj.) (*doksa tōi Theōi tōi X (adj.)*, praise to God, the X (adj.), also Example 3.1).

4. If the n-gram yields a positive frequency count, it is more likely to be the formula itself, with the (n+1)-gram being the formula accompanied by an element (e.g. an optional modifier) that frequently co-occurs, although the n-gram occurs more frequently as a standalone entity. For example, συν, θεω (with (the help of) God) occurs 56 times, and τελος, συν, θεω (the end, with (the help of) God) occurs 31 times. Based on the dataset refinement described above, the non-redundant frequency count of the bi-gram is 25. Thus, we can say that συν, θεω is the formulaic element, frequently but not exclusively paired with τελος in our subset.

5. If the n-gram count equals zero, both the n-gram and (n+1)-gram occur equally (i.e., only together). For example, the 4-gram ξενοι, χαιρουσιν, ιδειν, πατριδα occurs 10 times, and the 5-gram ωσπερ, ξενοι, χαιρουσιν, ιδειν, πατριδα also occurs 10 times. Thus, the 4-gram is deemed insignificant (non-redundant frequency count is set to 0) for further exploration as a stand-alone formula. In this case, the (n+1)-gram constitutes a formulaic sequence, and the n-gram is exclusively a part of it and, thus, not an independent entity.

6. If the n-gram presents a negative frequency count, this indicates that the n-gram occurs as part of one or more (n+1)-grams. This suggests that the n-gram is a common recurring pattern included in one or more formulaic (n+1)-grams, with other elements intervening. This is because the n-gram is captured in fixed order by default, but the subtraction procedure for eliminating redundancy considers the elements of the n- and (n+1)-grams in free order. In short, a negative frequency count reveals that the n-gram represents a formulaic element of lower hierarchy but is still related to the one represented by the (n+1)-gram. For example, the bi-gram χριστε, σωσον occurs twice, while

[7] e.g., καὶ οἱ *kai hoi* (and the), ἐν τῇ *en tēi* (in the), or τῶν ἐμῶν *tōn emōn* (of mine)

[8] cf. Kuiper (2009) '+nlcu', Non-Linguistic Conditions of Use)

the 3-gram χριστε, μου, σωσον presents a frequency count of 15. In this case, the bi-gram's non-redundant count is set to -13, and it is considered a related, less frequent variant of the 3-gram. For our purposes, these n-grams will be referred to as *component-formulae*, reserving the term *formula* for the (n+1)-grams in this context.

7. Formulae can be maximum one verse long based on our Method (4.4). This means that *component-formulae* are shorter, while multi-verse entities are here considered a compilation of different formulae and called *patterns*.

Through a systematic comparison of n-grams across the corpus, recurring patterns suggestive of standalone formulaicity were isolated and documented. That resulted in a list of formulaic material within our DBBE subset, the frequency of which we acquired in step 3 and by n-gram extraction.

5. Results

Due to space constraints, this section will primarily discuss select yet representative findings.

The analysis will start from a well-known and established formula which recurs in Byzantine manuscripts, commonly known as the 'ὥσπερ ξένοι formula' (Example 1). Based on our last dataset refinement (see 4.6) and the criteria outlined above (in 4.7), it is confirmed that it constitutes a multi-verse pattern consisting of verse-long formulae.

The ὥσπερ ξένοι pattern's initial part/formula appears in two primary recurring forms (Table 2, A), with a total count of 14 occurrences in non-identical epigrams. At this level of analysis, there is no need to consider the difference between χαίρουσι (verb) and χαίροντες (participle), both stemming from the verb χαίρω ('to rejoice'). Similarly, the pattern's second part/formula (B, total frequency 6) exhibits minor spelling variations in the adverb (οὕτως / οὕτω). Although the formula is typically represented in scholarship by these two verses (as seen in Example 1), our results (Table 2) showed that this represents only half the truth in book epigrams, as A and B do occur consecutively (AB, see co-occ1), albeit with a frequency as low as 3 in our subset.

Formulae C and D represent structures mirroring formula A of the pattern, thus enriching the ὥσπερ ξένοι simile structure (e.g. ACDB). Among these, while D occurs often, the formulaic structures C, differentiated only by the use of different verbs (εὑρίσκω and ὁράω, respectively), collectively amount to double the frequency of D. This indicates that in cases of creative elaboration, the parallel structure/formula C is preferred. This preference aligns with Treu (1977), who suggests that C was the prototypical form of the formula, possibly dating back to the Greco-Roman period. Its earliest attestation, however, is to be found in the 9th century (*Palatinus gr.* 44), with formula A dominating only from the 10th century onward (*Parisinus gr.* 781).

Interestingly, our data shows that formula A does not exclusively replace formula C; rather, they frequently co-occur in several 'Types'. More specifically, A is combined with the most frequently occurring parallel structure (i.e. C) nine times, four of which are subsequently followed by the second part of the pattern (i.e. B).

The next example serves as another representative instance of a formula; through it, various aspects of the extracted formulaic material will be illustrated.

Firstly, akin to this example, most formulae are linked to and highlight the strong ritual aspect inherent in the corpus. As indicated in row 1 of Table 3, the 2-gram σὺν Θεῷ (*syn Theōi*, with (the help of) God) appears 56 times across different 'Types', solidifying its status as a prominent element in the Byzantine book epigram scribal repertoire. Given the deeply ingrained Christian context of Byzantine book production, it is unsurprising to observe such elements with high frequencies, with this particular 2-gram ranking as the 3rd most frequent formulaic sequence.

Moving on, row 2 presents what appears to be a half-verse formula. Comprising five syllables, it serves as an ideal sequence to occupy the first half-verse, with a caesura occurring after the 5th syllable, typical of the 12-syllable verse, the most frequently employed meter in the corpus.

Lastly, rows 3 and 4 offer examples of what we previously referred to as an *open-slot formula*. Half of the 3-grams (row 2) present a complement, as evidenced by the results of the formulaic 4-grams (row 3). Along with the absence of this pattern in the formulaic 5-grams, this suggests the presence of an empty slot, allowing for the inclusion of non-specific material, provided it constitutes a noun with or without modifiers (e.g., τέλος σὺν θεῷ τῆς θεολόγου βίβλου, 'the end, with (the help of) God, of the theological book'). Notably, the two instances of the 4-gram formula here are grammatically identical (τέλος σὺν Θεῷ ARTICLE). However, until a reliable part-of-speech tagger for Byzantine Greek is developed (Swaelens et al., 2023), researchers must manually identify this grammatical similarity.

6. Conclusion & Future Research

In this study, we have presented a methodology for identifying and extracting formulaic sequences from a subset of the Database of Byzantine Book Epigrams. The systematic application of this methodology offered a quantitative perspective of the prevalence and characteristics of formulaic language in

N-gram	F[9]	
ὥσπερ ξένοι χαίρουσιν ἰδεῖν πατρίδα	10	A
ὥσπερ ξένοι χαίροντες ἰδεῖν πατρίδα	4	A
οὕτως καὶ οἱ γράφοντες ἰδεῖν βιβλίου τέλος	3	B
οὕτω καὶ οἱ γράφοντες ἰδεῖν βιβλίου τέλος	3	B
ἰδεῖν πατρίδα | οὕτως καὶ οἱ γράφοντες	3	co-occ1
καὶ οἱ θαλαττεύοντες εὑρεὶν λιμένα	10	C
καὶ οἱ θαλαττεύοντες ἰδεῖν λιμένα	3	C
πατρίδα | καὶ οἱ θαλαττεύοντες	9	co-occ2
λιμένα | οὕτως καὶ οἱ γράφοντες	4	co-occ3
καὶ οἱ στρατευόμενοι ἰδεῖν τὸ νῖκος	7	D

Table 2: The multi-verse ὥσπερ ξένοι pattern

N-gram	F
σὺν Θεῷ	56
τέλος σὺν Θεῷ*[10]	31
τέλος σὺν Θεῷ τῆς **X**	8
τέλος σὺν Θεῷ τοῦ **X**	7

Table 3: Ritual repertoire, half-verse and open-slot formulae

the Byzantine book epigram tradition. Our methodology provides a systematic framework for an initial step towards a comprehensive study of the formulaicity and creativity present in Byzantine book epigrams.

Nevertheless, our study is not without limitations. The methodology relies on computational analysis, which overlooks certain nuances or cultural contexts inherent in the corpus. Therefore, this paper proposes machine-assisted extraction of formulaic material as an initial step before engaging in linguistic and philological research. Additionally, the current focus on the DBBE subset limits the generalizability of our findings to the broader Byzantine epigram tradition, encompassing both book contexts and other mediums. Future research presents potential for further refinement of our methodology. Statistical analysis could aid in discerning patterns or collocations that form fixed sequences, distinguishing them from purely grammatical ones. Moreover, the creation of a reliable lemmatiser for Byzantine Greek could eliminate the distinctions between patterns that vary only in morphological details. Similarly, the development of a tool for addressing itacism would eliminate discrepancies arising from different orthographic representations. Lastly, we will also investigate automatic part-of-speech tagging for Byzantine Greek (Swaelens et al., 2023) to identify *open-slot formulae*.

7. Bibliographical References

References

Egbert Jan Bakker. 2005. *Pointing at the Past : From Formula to Performance in Homeric Poetics*. Cambridge : Harvard university. Centre for Hellenic studies.

Daphne Baratz. 2015. The repetitive structure in verse: A comparative study in homeric, south slavic, and ugaritic poetry. *Greek, Roman and Byzantine studies*, 55:1–24.

Klaas Bentein. 2023. A Typology of Variations in the Ancient Greek Epistolary Frame (I–III AD). In Georgios K. Giannakis, Panagiotis Filos, Emilio Crespo, and Jesús de la Villa, editors, *Classical Philology and Linguistics: Old Themes and New Perspectives*, pages 429–472. De Gruyter.

Floris Bernard and Kristoffel Demoen. 2019. Book epigrams. In Wolfram Hörandner, Andreas Rhoby, and Nikolaos Zagklas, editors, *A companion to Byzantine poetry*, volume 4 of *Brill's Companions to the Byzantine World*, pages 404–429. Brill.

Julie Boeten, Mark Janse, Klaas Bentein, and Ilse De Vos. 2021. *Byzantine Metre from the Margins : A Corpus-Based, Pragmatic Analysis of Medieval Book Epigrams*. Ph.D. thesis, Ghent.

Chiara Bozzone. 2010. New Perspectives on Formulaicity. In Stephanie Jamison, W., H.-G. Melchert, and Brent Vine, editors, *Proceedings of the 21st Annual UCLA Indo-European Conference*, pages 27–44. Hempen Verlag, Bremen.

Chiara Bozzone. 2014. *Constructions: A New Approach to Formularity, Discourse, and Syntax in Homer*. Ph.D. thesis, University of California, Los Angeles.

Andreas Buerki. 2016a. Automatic identification of formulaic sequences in (fairly) big data: practical introduction to a procedure. *Advances in Identifying Formulaic Sequences:A Methodological Workshop*.

Andreas Buerki. 2016b. Formulaic sequences: A drop in the ocean of constructions or something more significant? In Ian McKenzie and Martin A. Kayman, editors, *Formulaicity and Creativity in Language and Literature*, pages 15–36. Taylor & Francis 2018.

Silvie Cinková, Pavel Pecina, Petr Podveský, and Pavel Schlesinger. 2006. Semi-automatic building of Swedish collocation lexicon. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

Giovanni. Ciotti and Hang. Lin, editors. 2016. *Tracing Manuscripts in Time and Space Through Paratexts Perspectives from Paratexts*. De Gruyter.

Viviana Cárdenas Cortes. 2022. *Lingüística de corpus en español / The Routledge Handbook of Spanish Corpus Linguistics*, chapter Corpus del español y lenguaje formulaico. Routledge.

Gilles-Maurice de Schryver. 2008. Why Does Africa Need Sinclair? *International Journal of Lexicography*, 21(3):267–291.

Kaja Dobrovoljc. 2020. Identifying dictionary-relevant formulaic sequences in written and spoken corpora. *International Journal of Lexicography*, 33(4):417–442.

John Miles Foley. 2007. "Reading" Homer through Oral Tradition. *College Literature*, 34(2):1–28.

Gérard Garitte. 1962. Sur une formule des colophons de manuscrits grecs. *Collectanea Vaticana in honorem Anselmi M*, 1:359–390.

Gerard Genette. 1987. *Seuils*. Seuil, Paris.

Adele E. Goldberg. 2006. *Constructions at Work the Nature of Generalization in Language*. Oxford : Oxford University Press.

A.E. Goldberg. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Cognitive Theory of Language and Culture Series. University of Chicago Press.

Elizabeth Jeffreys and Michael Jeffreys. 1983. The Style of Byzantine Popular Poetry : Recent Work. *Harvard Ukrainian Studies*, 7:309–343.

Michael Jeffreys and Elizabeth Jeffreys. 1986. The Oral Background of Byzantine Popular Poetry. *Oral Tradition*, 1(3):504–547.

Michael J. Jeffreys. 1973. Formulas in the Chronicle of the Morea. *Dumbarton Oaks Papers*, 27:163–195.

A. D. Kominis. 1966. *To Byzantinikon Hieron Epigramma kai hoi Epigrammatopoioi*. Athens.

Koenraad Kuiper. 1996. *Smooth Talkers : The Linguistic Performance of Auctioneers and Sportscasters*. Mahwah (N.J.) : Erlbaum.

Koenraad Kuiper. 2000. On the Linguistic Properties of Formulaic Speech. *Oral Tradition*, 15(3):279–305.

Koenraad Kuiper. 2009. *Formulaic Genres*. Houndmills.

M.D. Lauxtermann, Österreichische Akademie der Wissenschaften. Kommission für Byzantinistik, and Universität Wien. Institut für Byzantinistik und Neogräzistik. 2003. *Byzantine Poetry from Pisides to Geometres: Epigrams in context*. Byzantine Poetry from Pisides to Geometres: Texts and Contexts. Verlag der Österreichischen Akademie der Wissenschaften, Wien.

Maria Letizia Lazzarini. 1976. *Le formule delle dediche votive nella Grecia arcaica*. Roma : Accademia nazionale dei Lincei.

Martti Leiwo. 2005. *Ancient Greece at the Turn of the Millennium. Recent Work and Future Perspectives. Proceedings of the Athens Symposium, 18-20 May 2001*, chapter Substandard Greek. Remarks from Mons Claudianus. Publications of the Canadian Institute in Greece, No. 4.

Albert B. Lord. 1960. *The singer of tales*. Harvard University Press.

Claudine Moulin, Iryna Gurevych, Natalia Filatkina, and Richard Eckart de Castilho. 2015. *Historical Corpora: Challenges and Perspectives*, chapter Analyzing Formulaic Patterns in Historical Corpora. Narr Publishing House.

Delphine Nachtergaele. 2015. *The Formulaic Language of the Greek Private Papyrus Letters*. Ph.D. thesis, Ghent University.

Rachele Ricceri, Klaas Bentein, Floris Bernard, Antoon Bronselaer, Els De Paermentier, Pieterjan De Potter, Guy De Tré, Ilse De Vos, Maxime Deforche, Kristoffel Demoen, Els Lefever, Anne-Sophie Rouckhout, and Colin Swaelens. 2023. The database of byzantine book epigrams project: Principles, challenges, opportunities. *Journal of Data Mining & Digital Humanities*.

Gijsbert Rutten and Marijke van der Wal. 2012. Functions of epistolary formulae in Dutch letters from the seventeenth and eighteenth centuries. *Journal of historical pragmatics*, 13(2):173–201.

Avedis K. Sanjian, editor. 1969. *Colophons of Armenian Manuscripts 1301-1480: A Source of Middle Eastern History*. Cambridge.

J. Sinclair. 1991. *Corpus, Concordance, Collocation*. Describing English language. Oxford University Press.

Merja Stenroos. 2018. *Scribal Repertoires in Egypt from the New Kingdom to the Early Islamic Period*, chapter From Scribal Repertoire to Text Community: The Challenge of Variable Writing Systems. Oxford University Press.

Colin Swaelens, Ilse De Vos, and Els Lefever. 2023. Linguistic annotation of byzantine book epigrams. *Language Resources and Evaluation*.

Mariken Teeuwen and Irene van Renswoude, editors. 2018. *The annotated book in the early Middle Ages : practices of reading and writing*. Turnhout : Brepols.

Kurt Treu. 1977. *Der Schreiber am Ziel. Zu den Versen* ῞Ωσπερ ξένοι χαίρουσιν... *und ähnlichen*, pages 473–492. Berlin: Akademie-Verlag.

Emmanuel van Elverdinghe. 2023. *Armenia and Byzantium without Borders: Mobility, Interactions, Responses*, chapter The Hand That Once Wrote ...': The Journey of a Colophon Formula from Greek into Armenian. Brill.

Evert van Emde Boas, Albert Rijksbaron, Luuk Huitink, and Mathieu de Bakker. 2019. *The Cambridge Grammar of Classical Greek*. Cambridge University Press.

Alison Wray. 2002. *Formulaic Language and the Lexicon*. Cambridge University Press.

Alison Wray. 2008. *Formulaic language: Pushing the boundaries*. Oxford Applied Linguistics. Oxford University Press.

How relevant is part-of-speech information to compute similarity between Greek verses in a graph database?

Colin Swaelens†, Maxime Deforche‡, Guy Detré‡, Ilse De Vos⋆, Els Lefever†
†Language and Translation Technology Team, Ghent University, Belgium
‡ Dpt. of Telecommunications and Information Processing, Ghent University, Belgium
⋆VAIA - Flanders AI Academy, Heverlee, Belgium
{colin.swaelens,maxime.deforche,guy.detre,els.lefever}@ugent.be, ilse.devos@kuleuven.be

Abstract

This paper presents the automatic linguistic analysis of the Database of Byzantine Book Epigrams (DBBE) on the one hand, and its representation and integration in a graph database on the other hand. Firstly, we provide a comprehensive description of the DBBE data we want to provide with a complete morphological analysis. The presented methodology explores the possibilities of fine-tuning the DBBErt transformer-based language model, which was trained on pre-Modern and Modern Greek. Secondly, the automatically annotated epigrams are integrated in a graph database, a new way to represent the relatedness of this entangled corpus. With the graph database, we can compute similarity between words, verses and epigrams. Given the scope of this paper, we computed a complete orthographic similarity between the verses, a similarity based on the automatically assigned part-of-speech information and a final similarity measure that combines both orthography and part-of-speech information. The results of these similarity measures provide scholars with new visual representations of relations between (parts of) texts, which is beneficial for new critical editions and commentaries.

Keywords: Ancient Language Processing, Graph Database, Similarity Search

1. Introduction

The traditional way of making a historical text accessible to the general public typically involves the production of a critical edition. Through a critical edition, an editor, viz. a philologist, presents their interpretation of what the original text (*Uhrtext*) likely was, drawing from the manuscripts that have survived over time. Beneath the main text of a critical edition, the apparatus displays all variants of a given word within the text, as found in the manuscripts. The greater the number of manuscripts included, the more *critical* the edition becomes. The question arises as to whether research would benefit from more dynamic systems in contrast to to the static system inherent to a critical edition. A more dynamic system could, for instance, store linguistic information for each word within the text, offering better insight into the variation of textual readings. In this paper, we propose such a dynamic system built upon a graph database framework, which facilitates the grouping of similar words, verses and even complete chunks of text. The similarity measure may rely solely on orthographic criteria, or it can take into account variation in spelling as well as flexible word order. Incorporating linguistic information, such as part-of-speech tags, morphological features, or semantic labels, enables the utilisation of the most fine-grained queries to identify related textual segments. Such a tool empowers philologists to make more robust and comprehensive critical editions as well as commentaries.

In this paper, we introduce the first version of this system, which incorporates various orthography-based similarity measures alongside automatically tagged part-of-speech information. The corpus we work with comprises Byzantine book epigrams, which are poems typically inscribed in the margins of manuscripts by the scribe of the manuscripts themselves. Editions of these book epigrams do exist (Rhoby, 2018), but the Database of Byzantine Book Epigrams (DBBE) (Ricceri et al., 2023) has the unique benefit of storing both the verbatim transcription of the epigrams as well as their edition-like variants, called *Occurrence* and *Type* respectively. As the DBBE *Occurrences* present the epigrams exactly as they appear in the manuscripts, they exhibit quite some inconsistencies, including variations in orthography, punctuation, and metre. This stems from the epigrams being predominantly autographs, a sharp contrast to classical texts that have been copied and edited over centuries.

2. Literature Review

The visual grouping tool presented in this paper integrates the focal points of this literature review: orthographic similarity and linguistic annotation of Greek.

2.1. Orthographic Similarity

Orthographic similarity measures seek to calculate a similarity score between two texts, purely based on the likeness between individual tokens or characters comprising the text, without considering contextual information or semantics. Character-based orthographic measures such as N-grams (Kondrak, 2005), Jaro(-Winkler) (Winkler, 1990; Jaro, 1995), and Damerau(-Levenshtein) (Damerau, 1964; Levenshtein, 1966) compute string similarity by comparing sequences of individual characters. Likewise, token-based similarity measures, like the Overlap Coefficient, the Cosine Similarity or the Jaccard Similarity (Jaccard, 1901; Gomaa et al., 2013) produce a similarity score by comparing between sets or sequences of complete tokens. Few techniques that combine both token- and character-based methods, have been investigated. These hybrid techniques ascertain the similarity between two tokens by considering the underlying character-based similarity score of those two tokens (Bronselaer and De Tré, 2009; Gali et al., 2019). Traditional orthographic methods typically aim to compute a single, comprehensive similarity score, without taking into account the underlying structural intricacies of the texts. However, when assessing the similarity among (the components of) Byzantine book epigrams, which constitute highly interconnected semi-structured texts, these methods prove inadequate.

Deforche et al. (2024) have proposed a new, innovative orthographic similarity measure: it supersedes the notion of simply merging character- and token-based measures and instead deals with texts in a more structured manner. This novel method breaks down texts into hierarchical discourse units, like words or verses, and, commencing from the smallest units, proceeds to compute similarities between all elements belonging to the same discourse unit. These hierarchical similarity calculations draw inspiration from the Damerau-Levenshtein distance (Damerau, 1964), and the computations for a specific discourse unit will integrate the precomputed similarity scores between the lower units of discourse. Furthermore, the hierarchical breakdown of texts, coupled with the similarity scores between the elements of each discourse unit, can be stored in a graph database (Angles and Gutierrez, 2008). By leveraging the advanced and/or visual querying capabilities of these databases, new methods and tools for exploring and analysing textual corpora can be devised. This hierarchical method has yielded promising results in computing orthographic similarities among (segments of) Byzantine book epigrams, where each epigram is represented by a hierarchical decomposition of tokens, verses, and complete texts (Deforche et al., 2023, 2024).

2.2. Part-of-Speech Tagging

Part-of-speech tagging involves assigning a part-of-speech label to each token in a text. While this task might be fundamental in natural language processing, it becomes non-trivial when applied to historical languages. The initial algorithms devised for part-of-speech tagging in Greek texts, combined a rule-based approach with a dictionary look-up (Packard, 1973; Crane, 1991). Given that the to-be-tagged text is edited to a classical standard, Crane's algorithm, Morpheus, remains competitive compared to more recent developments, such as RNN Tagger (Schmid, 2019). This neural-based part-of-speech tagger represents the first Greek-specific tagging algorithm introduced since Morpheus. In the three decades between Morpheus and RNN Tagger, existing part-of-speech taggers have been (re-)trained on classical Greek data, ranging from HMM-based (Halácsy et al., 2007) and statistical models (Bohnet and Nivre, 2012), over decision-tree based models (Schmid, 1994; Schmid and Laws, 2008) to Conditional Random Fields (CRF) (Müller et al., 2013).

When tagging morphologically rich languages like Greek, Latin, or Sanskrit, the part-of-speech tag is typically supplemented with the token's morphological features. In the case of Greek, the initial algorithms mentioned above (Packard, 1973; Crane, 1991) provided a complete morphological analysis in addition to their part-of-speech tag. None of those algorithms, however, disambiguate ambiguous word forms, which are quite common in Greek; instead, they provide all possible analyses of a word form. To illustrate, the Morpheus algorithm was unable to provide a single morphological analysis of 47.37% of our test set (cf. Section 3.1). Building upon the survey articles by Celano et al. (2016) and Keersmaekers (2019) which focused on classical and papyrological Greek respectively, Swaelens et al. (2023b) conducted a comparison between RNN Tagger and transformer-based part-of-speech taggers on unedited Byzantine Greek. Drawing inspiration from the exploratory research of Singh et al. (2021), they developed a pipeline that utilises contextualised token embeddings from the DBBErt model[1] as input for a bi-directional Long Short-Term Memory (LSTM) encoder and a CRF decoder, made available by the FLAIR framework (Akbik et al., 2019). As a second approach, they undertook fine-tuning of the contextualised token embeddings directly for part-of-speech tagging. This approach yielded results comparable to those achieved with the combination of a bi-LSTM encoder with a CRF decoder.

[1] `https://huggingface.co/colinswaelens/DBBErt`

3. Linguistic Annotation

3.1. Data

The majority of NLP techniques outlined in Section 2.2 are trained and evaluated on Classical Greek data sourced from editions. These editions are based on manuscripts, but any inconsistencies encountered are adjusted to fit a Classical Greek model. However, our focus lies in original, unedited texts which are gaining prominence thanks to the growing interest in optical character recognition (OCR) and handwritten text recognition (HTR) (Bhunia et al., 2021; Nockels et al., 2022; Retsinas et al., 2022). Regrettably, the available quantity of unedited Greek data containing linguistic annotation is currently insufficient to compile both a training and test set. At present, we have annotated a test set comprising approximately 10,000 tokens of unedited Byzantine Greek sourced from the DBBE *Occurrences*. We manually provided this test set with part-of-speech tags, morphological features, and lemmas. Further details are comprehensively reported by Swaelens et al. (2023b).

The training data used for the experiment that we present in Section 3.2, is a combination of PROIEL (Haug and Jøhndal, 2008), the Ancient Greek Dependency Treebanks (Celano, 2019; Bamman and Crane, 2011), the Gorman treebanks (Gorman, 2020), the texts provided by Trismegistos (Keersmaekers and Depauw, 2022), and the Pedalion trees (Keersmaekers et al., 2019). From these treebanks, we extracted the part-of-speech tag, morphological analysis, and lemma of each token. Lemmas are not yet taken into account for the experiments presented in this paper because the development of a lemmatiser for unedited Greek is still in progress (Swaelens et al., 2023a, 2024).

3.2. Method

Our initial objective is to offer a full morphological analysis of some 8,000 unedited Byzantine Greek tokens. The tag for this morphological analysis consists of nine slots, each corresponding to one of the following features, as put forward by the universal dependencies framework (Nivre et al., 2016): part-of-speech, person, number, tense, mood, voice, case, gender, and degree of comparison. Previous research adopted a two-step approach: initially predicting only the part-of-speech, followed by a second step where a single label encompassing all morphological features was predicted. Figure 1 depicts the results of two transformer-based approaches for both labelling part-of-speech and conducting morphological analysis (Swaelens et al., 2023b). These results are compared against a most-frequent-label baseline on the one hand, and the RNN Tagger on the other.

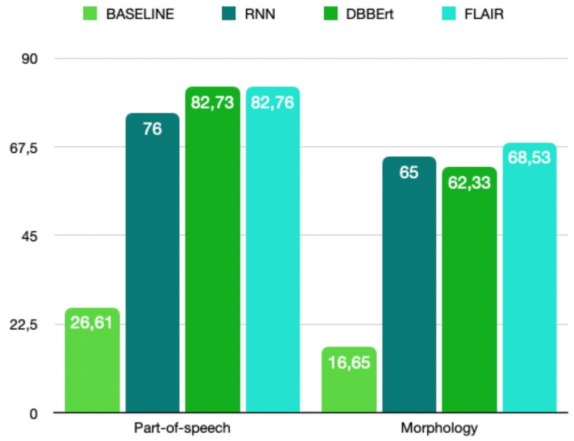


Figure 1: Results of existing transformer-based linguistic annotation of Byzantine Greek.

For the task of part-of-speech tagging they yield accuracy scores of 26.61% and 76.97% respectively. Fine-tuning the Greek transformer-based language model for part-of-speech tagging yielded an accuracy of 82.73%. The second approach, in which the transformer embeddings are processed in the FLAIR framework (cf. Section 2.2), in its turn, resulted in a tagging accuracy of 82.76%. For the task of morphological analysis, the baseline score is 16.65%, while analysis by the RNN Tagger resulted in 65.59%. With an accuracy of 62.33%, the DBBErt model fine-tuned on morphology performs 3 pp. less than the RNN Tagger. When the transformer embeddings are utilised within FLAIR, the output slightly outperforms the RNN Tagger by 3 pp., achieving an accuracy of 68.53%.

Previous research has highlighted that the drop in performance between part-of-speech labelling and morphological analysis may be attributed to the magnitude of the morphological label set. This label set comprises 1,057 possible labels, whereas the part-of-speech labels amount to 14. However, it is noteworthy that the training for both tasks is conducted on the same, relatively modest training set. Nevertheless, we aim to elevate the performance of the automatic morphological analysis. Therefore, both a more novel and a more traditional approach are trained and evaluated for this classification task.

3.2.1. Transformer-based Approach

In our first experiment, we fine-tuned the DBBErt model for each of the nine features outlined in Section 3.2. Except for the feature ‘part-of-speech’, our biggest label set counts only 9 labels, while the smallest comprises no more than 4. The accuracy of each classifier ranged from 82.73% for case to 96.24% for person. We have excluded the scores for degree of comparison, since the classifier labelled all tokens with ‘-’, which indicates this

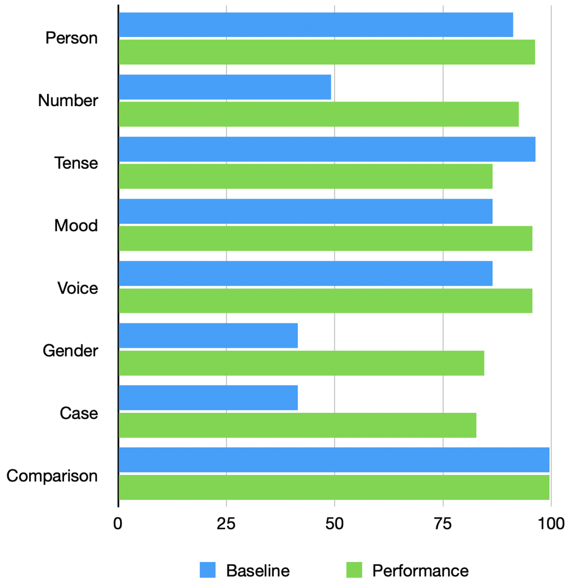


Figure 2: The performance of each of the fine-tuned DBBErt classifiers for the morphological features.

feature lacks labels. To verify that the other classifiers learned more than the one for the degree of comparison, Figure 2 displays the baseline of the most frequent label for each classifier alongside its performance. Despite these promising results, upon assembling the output of all classifiers, the accuracy of the combined label was only 58.4%. A recurring problem with the assembly method is the assignment of redundant features, such as nouns being labelled with 'present' for tense instead of '-'. To address this issue and prevent the assignment of redundant features, our second experiment employs a cascaded approach.

In this cascaded approach, the first step involves assigning the part-of-speech to a given token. Subsequently, only the classifiers of the features specific to the part-of-speech are employed to predict the label of that feature. For instance, if the part-of-speech is a noun, adjective, or pronoun, only the number, case, and gender classifiers predict a label, while the other features are automatically labelled as '-'. If the part-of-speech is a verb, we first determine the mood of the verb to predict the correct features. All verbs share the features voice, mood, and tense. An infinitive has no additional features, so the other slots are labelled as '-'. The indicative, imperative, subjunctive, and optative share the additional features person and number. The participle, on the other hand, has the additional features case, gender, and number. This cascaded approach, which combines rules with transformer-based classifiers, yielded an accuracy score of 58.29%. Contrary to our hypothesis, our cascaded approach did thus not outperform the assembly method.

These experiments suggest that transformer-based classifiers may not be suitable for the automatic morphological analysis of unedited Greek. Consequently, we explored a more traditional classification approach: support vector machines (SVM). These SVM classifiers are fed the transformer embeddings from the DBBErt model as input, a method known to be quite efficient for classification tasks (De Geyndt et al., 2022).

3.2.2. SVM

Typically, more traditional feature-based machine learning algorithms like SVM rely on manually crafted features, such as local context (preceding or next word) or linguistic information like part-of-speech. However, for this experiment, we generated transformer embeddings with the DBBErt_pos2024 model[2]. Since DBBErt_pos2024 is fine-tuned on part-of-speech tagging, these embeddings contain not only contextual information but also part-of-speech information. We adopted an approach similar to the one presented in Section 3.2.1.

Firstly, we trained an SVM classifier for the complete morphological tag, which resulted in an accuracy score of 39.43%. However, it classified practically all tokens as the punctuation label (u--------). When predicting the complete label at once, the SVM exhibited a drop in performance of almost 30 pp. compared to the best algorithm of Figure 1.

Secondly, we trained distinct SVM classifiers, similar to the approach with the nine transformer classifiers. This time, however, to conserve computational resources, we began with the morphological features of nouns, adjectives, and pronouns: case, gender, and number. These classifiers yielded accuracies of 75.34%, 90.94%, and 72.83% respectively. When these labels were assembled and redundant slots were assigned '-', this classification approach yielded an accuracy score of 58.07%. As the morphological features of the Greek verbal system are much more complex than those of the nominal system (more relevant features with more options), we decided not to train classifiers for the remaining morphological features for verbs, as they would likely perform even worse than the classifiers for nominal features.

4. Similarity Detection

Given the scope of this exploratory paper, the detection of similar texts is limited to identifying similar verses of unedited Byzantine Greek. The similarity detection relies not only on the orthographic similarity measures, as described in Section 2.1, but

[2] https://huggingface.co/colinswaelens/DBBErt_pos2024

also on the combination of these methods with automatically provided linguistic information. In an ideal scenario, the linguistic information consisted of both the part-of-speech tag and a full morphological analysis. However, since the tool for automated morphological analysis requires further improvement, the linguistic information integrated into the pipeline is limited to part-of-speech tags. The subsequent sections offer a detailed description of the workflow outlined in Figure 3.

4.1. Graph Database

In graph databases data are organised by means of graphs, unlike traditional relational databases where data are structured in tables (Angles and Gutierrez, 2008). Such graphs consist of nodes and relationships (or edges) connecting these nodes. Due to their structure, graph databases excel in handling highly interconnected data (Batra and Tyagi, 2012), making them an ideal tool for storing a large number of similarity relationships between texts, with each text represented by a node. Furthermore, graph database systems allow for advanced and visual analysis of the numerous interconnections between nodes, providing an ideal instrument for detecting and analysing similar texts.

For this paper, we have established such a graph database to store verses of Byzantine book epigrams. Before importing the texts into the graph, the verses undergo preprocessing to standardise them and reduce noise, thereby facilitating similarity calculation in later steps of the process. The preprocessing involves converting uppercase characters to lowercase and removing punctuation and diacritics. Subsequently, these preprocessed verses are stored in dedicated verse nodes in the graph. However, verses that maintain the exact same spelling after preprocessing, are stored in a single node.

Not only complete verses but also individual words are stored in the graph. Words are tokenised by splitting up the preprocessed verses based on white spaces, and like verses, these words are stored in dedicated word nodes, where identical words are – again – represented by a single node. Nodes representing words are also connected to the verse node in which they appear. These relationships include information about both the rank and the part-of-speech tag of that word within the connected verse.

4.2. Method

Utilising the preprocessed verses and tokens already stored in the graph database, our objective is to compute three similarity scores between each pair of verse nodes: orthographic similarity, part-of-speech-based similarity, and a combination of both. The outcome of these similarity calculations is a score between 0 and 1, denoting the degree of similarity between two verses based on the specific similarity measure employed. A score of 0 indicates complete dissimilarity between two verses, whereas a score of 1 signifies complete similarity. The remainder of this section provides a succinct description of these three similarity measures.

4.2.1. Orthographic Similarity

The orthographic similarity between verse nodes is determined by employing an implementation of Deforche et al. (2024), utilising the default parameters of the algorithm[3]. This similarity measure firstly calculates the similarity between all word pairs, then utilises these word-level similarities to ascertain the similarity between all verse pairs. The process of determining the similarity between two words begins by computing the Damerau-Levenshtein edit distance (Damerau, 1964), which represents the minimal cost required to transform one word into another using one of the four supported edit operations: insertion, deletion, replacement of a single character, or the transposition of two consecutive characters. In this paper, we assume the cost of all mentioned edit operations to be equal to 1. The word-level similarity score is then obtained by dividing the resulting edit distance by the length (in characters) of the longest of the two words and subtracting this number from 1. In the case of Byzantine texts, this word-level similarity is computed without penalising either the itacism[4] nor diacritics. This means that, for example, the similarity between ξένοι and ξενη is 1, indicating that these words are treated as identical.

Next, a similar process is repeated to calculate the orthographic similarity scores between all pairs of verse nodes. In this case, the edit distance is calculated between two verses using the same four edit operations, but rather than considering individual characters, entire words are taken into account. Once again, all edit operations are presumed to have a cost of 1, except for the replacement operation between two words. In the case of replacements, the cost equals the dissimilarity between the word and its potential replacement, which can be determined by subtracting the precomputed similarity between those words from 1. Lastly, the edit distance between two verses needs to be converted into a similarity score. This is accomplished by dividing the resulting edit distance by the length (in words) of the longest verse and subtracting this number from 1. The resulting orthographic similarity score between two verse nodes is stored in the

[3] https://github.com/MaximeDeforche/DBBESimilarity

[4] The itacism is a phonetic shift of ει, η, ι, οι, υ into [i].

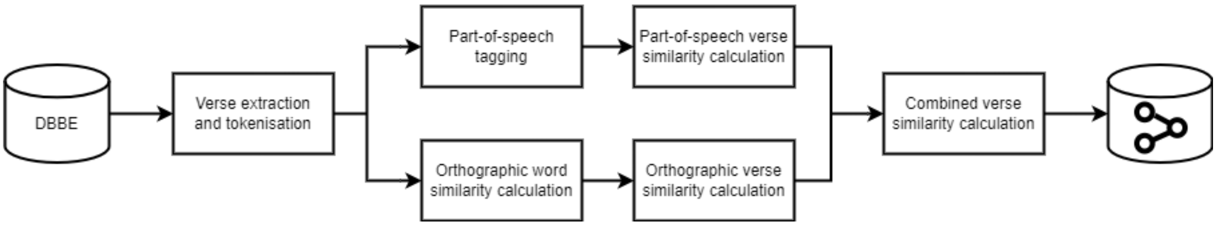

DBBE
Verse extraction and tokenisation
Part-of-speech tagging
Orthographic word similarity calculation
Part-of-speech verse similarity calculation
Orthographic verse similarity calculation
Combined verse similarity calculation

Figure 3: Workflow from relational database with plain text to a linguistically annotated graph database, including similarity scores between texts.

graph database by means of a relationship between those two nodes.

Further details on the implementation and customisation options of this orthographic similarity measure are comprehensively reported by Deforche et al. (2024).

4.2.2. Part-of-speech Similarity

Next, we compute a similarity measure based on the part-of-speech tags assigned to each word. For this similarity measure, we draw inspiration once again from the Damerau-Levenshtein edit distance (Damerau, 1964) to compute a part-of-speech similarity score between all verse nodes. For each verse node, we concatenate the part-of-speech tags of all words in a verse into a single string, maintaining the same order as the appearance of the words in that verse. Subsequently, the edit distance between these strings is determined by calculating the minimal cost required to transform one part-of-speech representation of a verse into the other. In this paper, the supported edit operations all have a cost of 1 and include the insertion, deletion, and replacement of a single part-of-speech tag, as well as the transposition between two consecutive part-of-speech tags. The resulting edit distance is then transformed into a similarity score by dividing it by the length (in words) of the longest verse and subtracting this result from 1. Finally, the resulting part-of-speech similarity score is stored in a similarity relationship connecting the two verse nodes for which this similarity is calculated.

As an example, we consider two verses that are represented by the part-of-speech tags of each word they consist of. The first verse consists of the tags: adverb (`d`), adjective (`a`), verb (`v`), noun (`n`) and verb (`v`), and the second verse of the tags: adverb (`d`), interjection (`i`), verb (`v`), verb (`v`), noun (`n`) and noun (`n`). First, the edit distance between `davnv` and `divvnn`, which are the concatenated part-of-speech tags of both verses, is determined. The edit operations to transform the concatenated tags from one verse into the other are visualised by Figure 4 and consist of a replacement (orange), a transposition (crossing arrows), and a insertion/deletion (green/red), resulting in a total edit distance of 3. Using this edit distance, the part-of-speech similarity between the verses is calculated using the method described above and results in similarity score of 0.5.

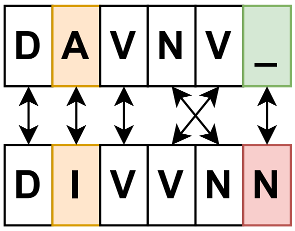


Figure 4: Edit operations between two concatenations of part-of-speech tags.

4.2.3. Combined Similarity

As a third and final measure, we aim to compute a similarity score between each pair of verse nodes that considers both the orthographic information and the automatically provided linguistic information. We calculate this score by averaging the orthographic and part-of-speech similarities already determined for each pair of verse nodes. In future research, we plan to explore more advanced and customisable options like the Ordered Weighted Average (OWA) operators (Yager, 1988; Yager and Kacprzyk, 2012) or the Logic Scoring of Preference (LSP) method (Dujmovic, 2018). This combined measure results in a balanced similarity score that considers both orthographic and part-of-speech similarities.

In a theoretical illustration, let us consider that the orthographic similarity between two verses is 0.6, while the part-of-speech similarity is to 0.8. Through the amalgamation of these two scores, we arrive at a combined similarity score of 0.7.

In parallel with the other similarity scores, these results are stored on the relationship between verse nodes in the graph database, allowing us to analyse the relations between all verses based on this hybrid similarity measure.

4.2.4. Visual Grouping of Verse Nodes

Upon computing the similarity scores between each pair of verse nodes, we harness the querying capabilities of the graph database to identify and thoroughly analyse verses that exhibit (dis)similarities. Utilising a specified similarity threshold, the graph database can be queried to reveal all verse nodes and their associated similarity relationships of a specific similarity measure scoring equal to or exceeding the specified threshold. Through visual representation of such query results, we observe the emergence of groups of verse nodes that demonstrate at least the specified level of similarity according to the chosen similarity measure. Opting for a high similarity threshold yields numerous groups of highly similar texts, whereas a lower similarity threshold produces fewer groups of texts with lower degrees of similarity. Although initially counterintuitive, selecting a lower similarity threshold can be interesting, particularly when examining texts rife with spelling variations or orthographic inconsistencies, such as Byzantine book epigrams.

The ability to select the similarity measure and threshold provides researchers with the flexibility to analyse texts in myriad ways. The similarity measures outlined in this paper offer the capability to visually identify similar verses based on their orthographic properties, linguistic information, or a blend of both.

5. Case Study

5.1. Data

For our case study, we will compute similarities between verses linked to *Types* 2148, 2150, and 4245 from the DBBE (Demoen et al., 2023). These *types* group 154 DBBE *occurrences*, resulting in a set of 410 verses. Given that identical verses are stored only once, this set is stored as 286 unique nodes in the graph database. No duplicate values are stored. Among these verses, 1a and 1c are shared across all three Types. Presumably, the number of verses gave rise to three distinct Types. The *occurrences* grouped under Type 2150, for instance, all comprise three verses, whereas Type 2148 encompasses *occurrences* consisting of only two verses; Conversely, Type 4245 links *occurrences* totalling six verses.

(1) a. Ὥσπερ ξένοι χαίρουσιν ἰδεῖν πατρίδα,
Hōsper xenoi chairoysin idein patrida,
Just like travellers rejoice by seing their homeland,

b. καὶ οἱ θαλαττεύοντες εὑρεῖν λιμένα,
kai hoi thalatteuontes eurein limena,
and sailors by finding a harbour,

c. οὕτως καὶ οἱ γράφοντες βιβλίου τέλος.
houtōs kai hoi grafontes bibliou telos.
so do scribes at the end of a book.
DBBE Type 2150

5.2. Orthographic Similarity

To showcase the capabilities of this dynamic system, we provide a visual render of verse grouping based on orthographic similarity exceeding 85% (Figure 5). The computation of this similarity measure involves two main steps. First the similarity between two words is computed without penalising either the itacism nor diacritics, as they appear arbitrary throughout the corpus (cf. Section 4.2.1). Then the similarity score of the verses is computed by combining the word similarity scores.

The group highlighted within the yellow frame in Figure 5, represents variants of verse 1a. This visual shows minimal outliers, indicating a high level of similarity between the verses. Notably, the word that causes most ‘dissimilarity’ is the third word of verse 1a, χαίρουσιν. Despite not penalising the itacism, the participle χαίροντες still displays a 55% similarity to the indicative χαίρουσιν, accounting for one-fifth of the verse’s overall similarity score. Verses within the blue frame are variants of Example 1a differing only in the use of the infinitive βλέπειν *blepein* (to look at) instead of ἰδεῖν *idein* (to see). Although semantically nearly identical, the variant using βλέπειν shows no similarity with the majority using ἰδεῖν.

The red frame encompasses verses like Example 1b. Surprisingly, 4 of the 43 verse variants contain a participle of the word κινδυνεύω *kinduneuō* (run risk) instead of the expected θαλαττεύω thalatteō (to be at sea). Despite them being unrelated, the similarity between these two participles is still 54%, which again accounts for one fifth of the verse similarity.

Verses grouped in the orange frame represent Example 1c. However, this group consists of two distinct parts connected by what we would call *bridge verses*. The left part lacks the verb ἰδεῖν preceding βιβλίου τέλος *bibliou telos* (the end of the book), including Example 1c. The right part, on the other hand, does have ἰδεῖν before βιβλίου τέλος. Additionally, this group has a variant that is not linked with this similarity measure: the green group. These variants do not display as subject the more common nominative οἱ γράφοντες *hoi grafontes* (the writers) as in Example 1c, but use instead a dative construction with τοῖς γράφοντοις *tois grafontois* (to the writers). Figure 6 provides a detailed visual of the differences between the dative construction of the verse variant on the right and the nominative construction of the verse variant on the left . The orthographic dissimilarity of both the article and the noun results in an orthographic similarity of 81.5% between the two verses.

The pink frame encompasses nine verse variants all counting more than three verses. The structure of the sentence follows that of Example 1b: καὶ *kai* (and) [placeholder] εὑρεῖν *heurein* (finds) [placeholder]. The first of the two placeholders is either a noun or a participle, wihle the second is most often a noun. If the only difference within one verse is the use of a participle of a different verb, as in the κινδυνεύω/θαλαττεύω example supra, the similarity score is still quite high. In these verses, for example, the second placeholder is filled with τὸ κέρδος *to kerdos* (profit), λιμένα *limena* (harbour) or νήκος *nèkos* (victory). These last two, display 0% and 33% similarity respectively to τὸ κέρδος, and 0% to each other. Combined with the dissimilarity in the first placeholder, results in these verse variants being grouped separately for this similarity measure.

The remaining verse variants will not be elaborated upon as these verses are not connected to more than two other verse variants. Most of them are incomplete verses due to lacunae.

It is important to keep in mind that Figure 5 provides a static representation, reflecting groups with a similarity score equal to or higher than 85%. However, the underlying system is dynamic, allowing adjustments to the similarity threshold which can be set lower or higher, and considerations for the itacism or other phonetic changes which can or cannot be penalised.

5.3. Implementation Part-of-Speech

This system could become even more dynamic with the implementation of linguistic annotation. Depending on your query, linguistic annotation could either refine search results by limiting them to specific parts-of-speech within verses, or, on the other hand, it could broaden the scope to include verses that display similarity based solely on part-of-speech information. As discussed in Section 3.2.2, in this paper the linguistic annotation is restricted to automatically labelled part-of-speech tags.

Once the similarity scores, as described in Section 4.2.3, are computed between verses in our dataset, the results are visualised in Figure 7. Notably, there are fewer verses that do not belong to any group compared to Figure 5. Another observation is the absence of verses from the green group in Figure 5. This is because the combination of part-of-speech information and orthography in a single similarity measure mitigates orthographic dissimilarities caused by the dative suffix resulting in the inclusion of those verses in the orange group. In Figure 6, the edge between the yellow verse nodes not only displays the orthographic similarity (81.5%) but also their combined similarity (90.7%), based on the part-of-speech labels visible on the edges between the yellow verse nodes and the green word nodes, representing their part-of-speech within that specific verse.

Similarly, one might anticipate the variants of Example 1a within the blue frame to integrate into the yellow group. However, despite the addition of part-of-speech information, these variants remain isolated. This suggests that part-of-speech information alone does not offset the penalisation of orthography and word order. Notably, the verses in the yellow group end with ἰδεῖν πατρίδα *idein patrida*, while those in the blue group end with πατρίδα βλέπειν *patrida blepein*.

6. Conclusion

We set out to explore the potential of a dynamic tool to assist scholars in their philological research endeavours. Our system operates in two main parts: first, the data is annotated with linguistic information; subsequently, users can select a similarity measure and define a threshold for similarity computation within the graph database. Currently, the linguistic information is limited to the automatic assignment of part-of-speech tags. The similarity measures presented include a purely orthographic measure, one based solely on part-of-speech, and a combined measure that integrates both aspects. Users have the flexibility to adjust the similarity measure and its threshold, tailoring the results to be either broad (with a lower similarity threshold) or specific (with a higher similarity threshold). With sufficient data in the graph database, scholars can uncover new relevant text segments to incorporate into their analysis or discover allusions to other authors for commentary purposes.

In future work, we plan to expand the relaxation rules of the itacism to include other phonetic changes in Byzantine Greek. We will also implement automatic morphological analysis, resulting in additional combined similarity measures. Furthermore, our focus will extend from orthographic to semantic similarity measures, exploring how these methods can be both flexibly and effectively combined in a manner that is specific to the field of study. We anticipate close collaborations with philologists to conceptualise a demo that will make this technology accessible to to the wider academic community.

7. Bibliographical References

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Renzo Angles and Claudio Gutierrez. 2008. Survey of graph database models. *ACM Computing Surveys (CSUR)*, 40(1):1–39.

David Bamman and Gregory Crane. 2011. The ancient greek and latin dependency treebanks. In *Language Technology for Cultural Heritage*, pages 79–98, Berlin, Heidelberg. Springer Berlin Heidelberg.

Shalini Batra and Charu Tyagi. 2012. Comparative analysis of relational and graph databases. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(2):509–512.

Ayan Kumar Bhunia, Shuvozit Ghose, Amandeep Kumar, Pinaki Nath Chowdhury, Aneeshan Sain, and Yi-Zhe Song. 2021. Metahtr: Towards writer-adaptive handwritten text recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15830–15839.

Bernd Bohnet and Joakim Nivre. 2012. A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1455–1465, Jeju Island, Korea. Association for Computational Linguistics.

Antoon Bronselaer and Guy De Tré. 2009. A possibilistic approach to string comparison. *IEEE Transactions on Fuzzy systems*, 17(1):208–223.

Giuseppe G. A. Celano, Gregory Crane, and Saeed Majidi. 2016. Part of speech tagging for ancient greek. *Open Linguistics*, 2(1).

Giuseppe GA Celano. 2019. The dependency treebanks for ancient greek and latin. *Digital Classical Philology*, page 279.

Gregory Crane. 1991. Generating and parsing classical greek. *Literary and Linguistic Computing*, 6(4):243–245.

Fred J Damerau. 1964. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3):171–176.

Ellen De Geyndt, Orphée De Clercq, Cynthia Van Hee, Els Lefever, Pranaydeep Singh, Olivier Parent, and Veronique Hoste. 2022. Sentemo : a multilingual adaptive platform for aspect-based sentiment and emotion analysis. In *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 51–61. Association for Computational Linguistics (ACL).

Maxime Deforche, Ilse De Vos, Antoon Bronselaer, and Guy De Tré. 2023. An orthographic similarity measure for graph-based text representations. In *Flexible Query Answering Systems*, pages 206–218, Cham. Springer Nature Switzerland.

Maxime Deforche, Ilse De Vos, Antoon Bronselaer, and Guy De Tré. 2024. A hierarchical orthographic similarity measure for interconnected texts represented by graphs. *Applied Sciences*, 14(4).

Kristoffel Demoen, Gilbert Bentein, Klaas Bentein, Floris Bernard, Julián Bértola, Julie Boeten, Mathijs Clement, Cristina Cocola, Eline Daveloose, Sien De Groot, Pieterjan De Potter, Ilse De Vos, Krystina Kubina, Hanne Lauwers, Paulien Lemay, Renaat Meesters, Marjolein Morbé, Delphine Nachtergaele, Marthe Nemegeer, Joachim Nielandt, Mace Ojala, Lisa-Lou Pechillon, Raf Praet, Rachele Ricceri, Anne-Sophie Rouckhout, Jeroen Schepens, Febe Schollaert, Lev Shadrin, Nina Sietis, Dimitrios Skrekas, Colin Swaelens, Maria Tomadaki, Sarah-Helena Van den Brande, Merel Van Nieuwerburgh, Lotte Van Olmen, Noor Vanhoe, and Nina Vanhoutte. 2023. Database of byzantine book epigrams.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jozo Dujmovic. 2018. *Soft computing evaluation logic: The LSP decision method and its applications*. John Wiley & Sons.

Najlah Gali, Radu Mariescu-Istodor, Damien Hostettler, and Pasi Fränti. 2019. Framework for syntactic string similarity measures. *Expert Systems with Applications*, 129:169–185.

Wael H Gomaa, Aly A Fahmy, et al. 2013. A survey of text similarity approaches. *international journal of Computer Applications*, 68(13):13–18.

Vanessa B Gorman. 2020. Dependency treebanks of ancient greek prose. *Journal of Open Humanities Data*, 6(1).

Péter Halácsy, Andras Kornai, and Csaba Oravecz. 2007. Hunpos: an open source trigram tagger. *Proceedings of the 45th Annual Meeting of the*

Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, pages 209–212.

Dag TT Haug and Marius Jøhndal. 2008. Creating a parallel treebank of the old indo-european bible translations. In *Proceedings of the second workshop on language technology for cultural heritage data (LaTeCH 2008)*, pages 27–34.

Paul Jaccard. 1901. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bull Soc Vaudoise Sci Nat*, 37:547–579.

Matthew A Jaro. 1995. Probabilistic linkage of large public health data files. *Statistics in medicine*, 14(5-7):491–498.

Alek Keersmaekers. 2019. Creating a richly annotated corpus of papyrological Greek: The possibilities of natural language processing approaches to a highly inflected historical language. *Digital Scholarship in the Humanities*, 35(1):67–82.

Alek Keersmaekers and Mark Depauw. 2022. Bringing together linguistics and social history in automated text analysis of greek papyri. *Digital Classics III: Re-Thinking Text Analysis (Classics@)*.

Alek Keersmaekers, Wouter Mercelis, Colin Swaelens, and Toon Van Hal. 2019. Creating, enriching and valorizing treebanks of Ancient Greek. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 109–117, Paris, France. Association for Computational Linguistics.

Grzegorz Kondrak. 2005. N-gram similarity and distance. In *International symposium on string processing and information retrieval*, pages 115–126. Springer.

Vladimir I et al. Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Soviet Union.

Thomas Müller, Helmut Schmid, and Hinrich Schütze. 2013. Efficient higher-order CRFs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332, Seattle, Washington, USA. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

Joe Nockels, Paul Gooding, Sarah Ames, and Melissa Terras. 2022. Understanding the application of handwritten text recognition technology in heritage contexts: a systematic review of transkribus in published research. *Archival Science*, 22(3):367–392.

David W. Packard. 1973. Computer-assisted morphological analysis of Ancient Greek. In *COLING 1973 Volume 2: Computational And Mathematical Linguistics: Proceedings of the International Conference on Computational Linguistics*.

George Retsinas, Giorgos Sfikas, Basilis Gatos, and Christophoros Nikou. 2022. Best practices for a handwritten text recognition system. In *International Workshop on Document Analysis Systems*, pages 247–259. Springer.

Andreas Rhoby. 2018. *Ausgewählte byzantinische Epigramme in illuminierten Handschriften*. Verlag der österreichischen Akademie der Wissenschaften, Wien.

Rachele Ricceri, Klaas Bentein, Floris Bernard, Antoon Bronselaer, Els De Paermentier, Pieterjan De Potter, Guy De Tré, Ilse De Vos, Maxime Deforche, Kristoffel Demoen, Els Lefever, Anne-Sophie Rouckhout, and Colin Swaelens. 2023. The database of byzantine book epigrams project: Principles, challenges, opportunities. *Journal of Data Mining & Digital Humanities*.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK.

Helmut Schmid. 2019. Deep learning-based morphological taggers and lemmatizers for annotating historical texts. In *Proceedings of the 3rd international conference on digital access to textual cultural heritage*, pages 133–137.

Helmut Schmid and Florian Laws. 2008. Estimation of conditional probabilities with decision trees and an application to fine-grained pos tagging. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, page 777–784, USA. Association for Computational Linguistics.

Pranaydeep Singh, Gorik Rutten, and Els Lefever. 2021. A pilot study for bert language modelling and morphological analysis for ancient and medieval greek. In *The 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, co-located with EMNLP 2021*, pages 128–137. Association for Computational Linguistics.

Colin Swaelens, Ilse De Vos, and Els Lefever. 2023a. Evaluating existing lemmatisers on unedited byzantine Greek poetry. In *Proceedings of the Ancient Language Processing Workshop*, pages 111–116, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Colin Swaelens, Ilse De Vos, and Els Lefever. 2023b. Linguistic annotation of byzantine book epigrams. *Language Resources and Evaluation*.

Colin Swaelens, Ilse De Vos, and Els and Lefever. 2024. Lemmatisation of medieval greek: Against the limits of transformers' capabilities? In *Proceedings of the Fourteenth Language Resources and Evaluation Conference*, Turin, Italy. European Language Resources Association.

William E Winkler. 1990. *String comparator metrics and enhanced decision rules in the Fellegi-Sunter model of record linkage.* ERIC.

Ronald R. Yager. 1988. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):183–190.

Ronald R. Yager and Janusz Kacprzyk. 2012. *The ordered weighted averaging operators: theory and applications*. Springer Science & Business Media.

Appendices: Figures

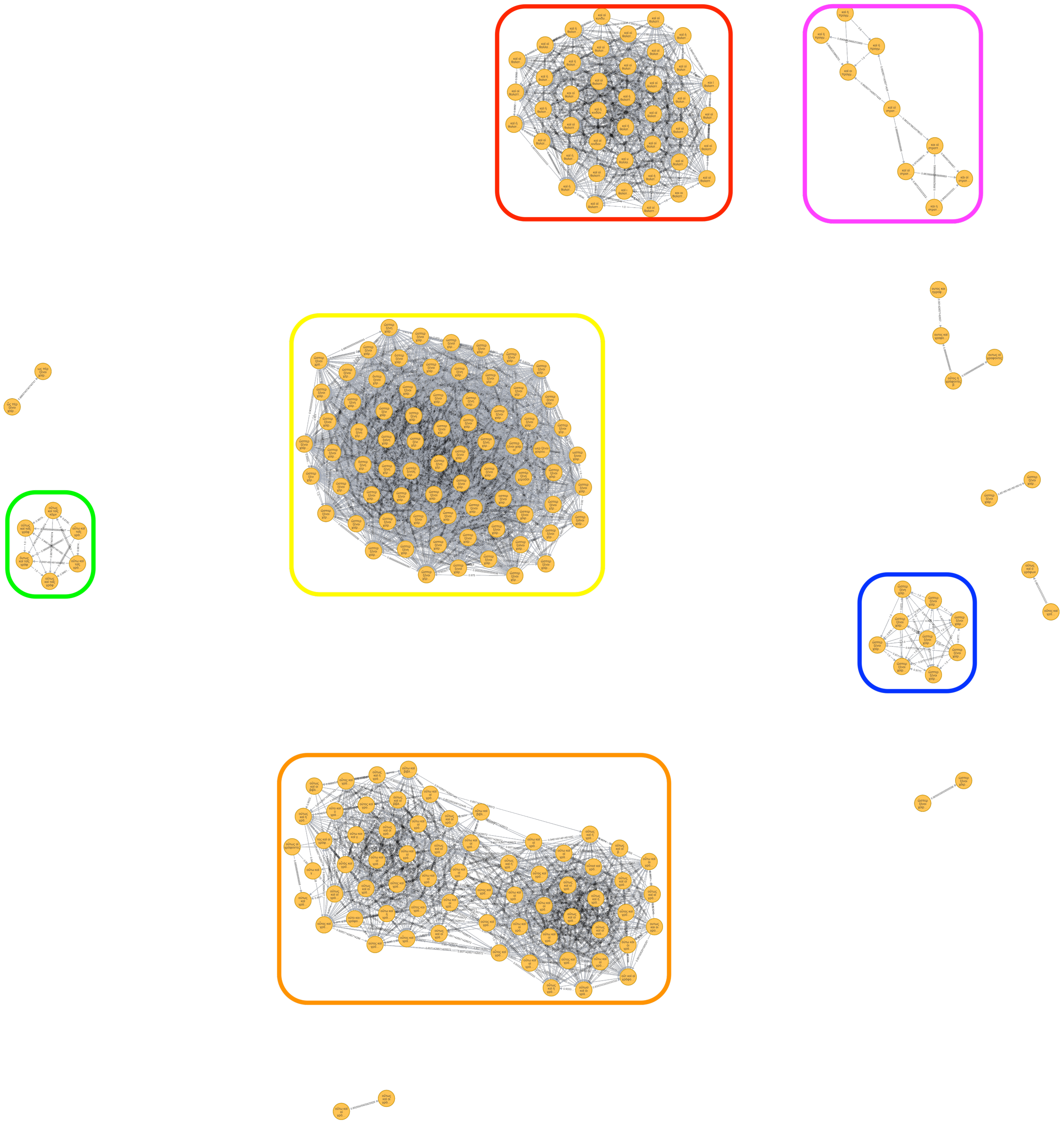


Figure 5: Orthographic similarity between verses

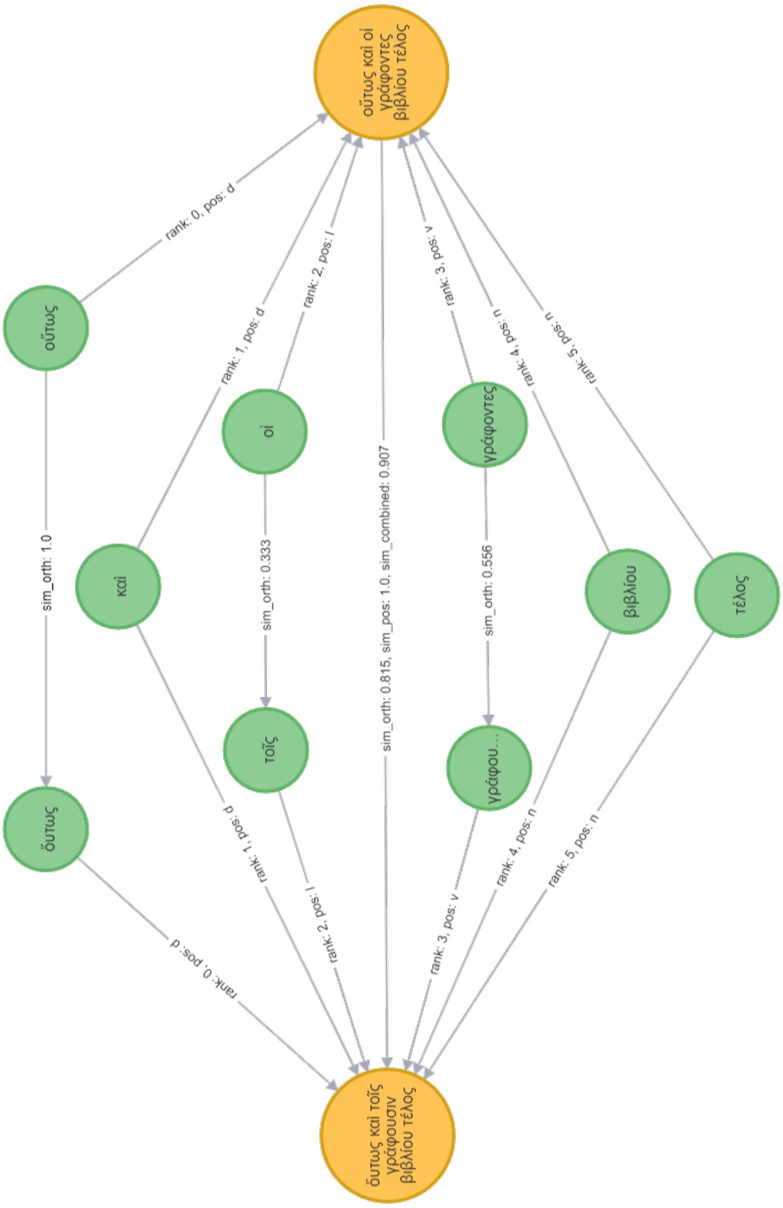


Figure 6: Detailed figure of verse variants of Example 1c: left with a dative construction, right with a nominative construction.

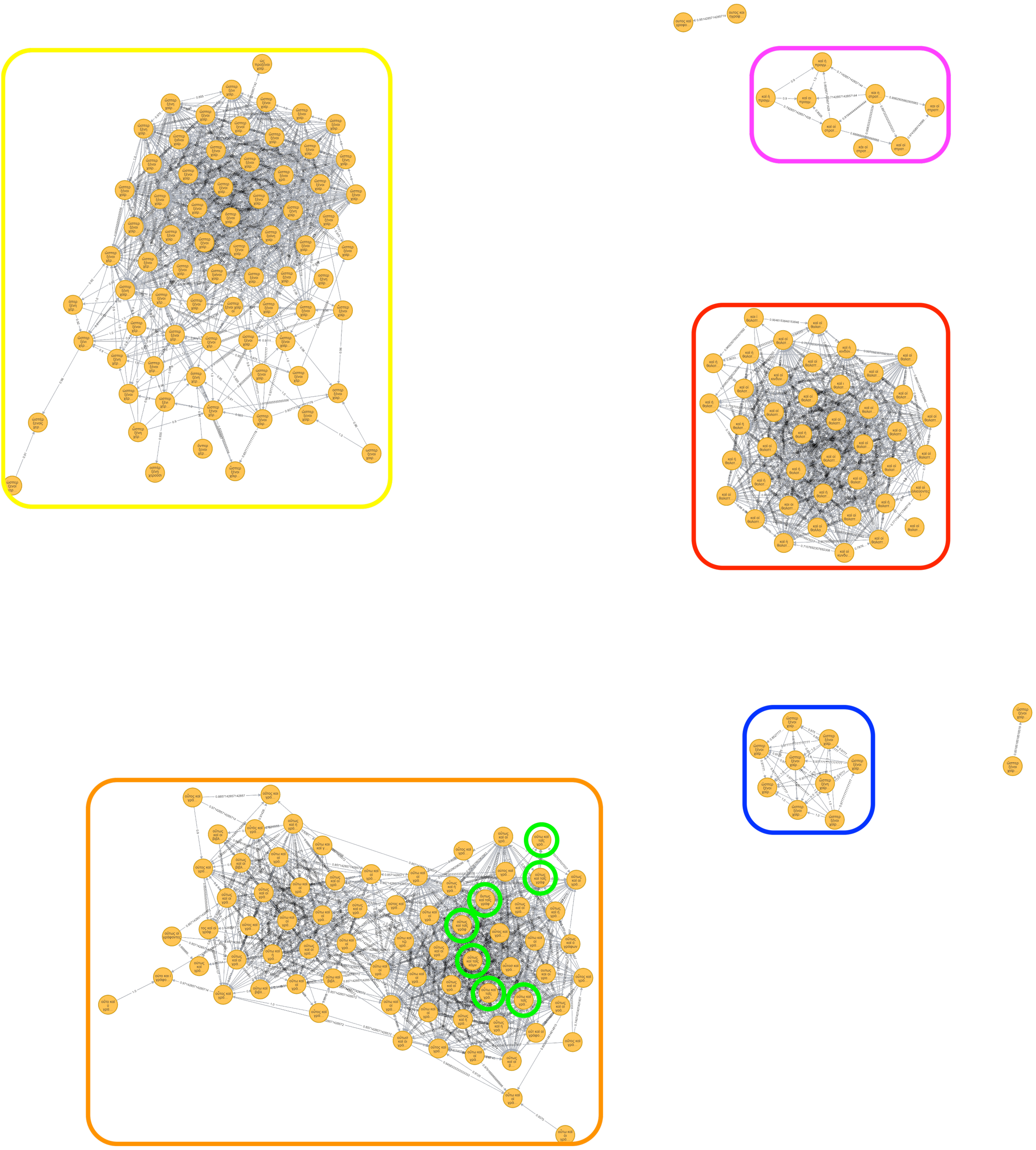


Figure 7: Bridge verses of orange group

Viability of Automatic Lexical Semantic Change Detection on a Diachronic Corpus of Literary Ancient Greek

Silvia Stopponi, Saskia Peels-Matthey, Malvina Nissim
Center for Language and Cognition, University of Groningen
Postbus 716, 9700 AS Groningen, The Netherlands
s.stopponi@rug.nl, s.peels@rug.nl, m.nissim@rug.nl

Abstract

We apply two measures of lexical semantic change detection to Word2Vec embeddings trained on a diachronic corpus of literary Ancient Greek texts. The two measures are *Vector Coherence*, based on the comparison between vectors of the same word in different time periods, and the J, based on the Jaccard coefficient, which quantifies the overlap between the k nearest neighbours in each possible combination of time slices. Through the analysis of the most stable and unstable words detected with both measures, we show that the two measures are effective at finding non-changed words, while Vector Coherence seems to be more reliable than J at detecting changed words. Still, low J could indicate a real semantic change when the same word also has a low Vector Coherence. For both measures, the detection of changed words is hampered by the presence of lemmatization errors in the training corpus.

Keywords: semantic change detection, Ancient Greek, language modelling, ancient language

1. Introduction

Changes in word meaning across time are of particular interest to linguists and to all scholars doing research about a specific culture, as semantic changes often reflect cultural and societal changes. Since the establishment of word embeddings as a computational tool to study word semantics (Mikolov et al., 2013), their application to study meaning diachronically has also rapidly emerged.

The core focus of such methods has been on modern languages, for many of which sufficient amounts of textual data exist to train word embedding models, and for which evaluation and analysis can be carried out by native speakers.

The picture is starkly different for ancient languages, since native speakers are lacking and our knowledge typically relies on a written corpus, limited in size, without the possibility of a substantial increase. Indeed, the study of word meaning for ancient languages has been generally carried out with the 'philological method', i.e. the manual examination of the occurrences of target word in context. Tracking meaning change then involves comparing (by hand) word occurrences from different time frames, extracted from a diachronic corpus.

Clearly, this approach to studying semantic change is an extremely time consuming process and does not leverage the potential of computational methods. At the same time, it is not obvious that such methods, and more specifically word embeddings, can be fruitfully used for ancient languages. To this end, we explore the viability of automatic lexical semantic change detection for Ancient Greek[1] with word embeddings testing two different measures of change. We assess if and how these measures can assist and complement the philological method to study lexical semantic change in Ancient Greek, highlighting their potential, their limitations, and possible solutions.

2. Challenges in Lexical Semantic Change for Ancient Greek

A general challenge of automatic lexical semantic change (LSC) detection is to discriminate real semantic changes from the effect of other factors present in the data. Systems for LSC detection based on language models detect words which underwent a change in usage, because they co-occur with different words in different time slices of the corpus (diachronic corpora are generally divided into time slices when performing LSC detection). However, differences in word usage can depend on several factors, such as corpus composition (e.g., different genres, topics or types of text unevenly distributed through time, as is the case for the Ancient Greek corpus[2]), and it is not necessarily a symptom of meaning change. For this reason, Gonen et al. (2020, 539) talk about 'usage change' instead of 'semantic change', and stress that *detected* words are just possible candidates for semantic change, and their status has to be manually verified. Indeed, with 'changed words' in this paper, we mainly refer to items detected by the used metrics, but

[1] Our corpus, the Diorisis Ancient Greek Corpus (Vatri and McGillivray, 2018), includes texts from the Homeric poems to 500 CE.

[2] For example, the Archaic portion of the corpus mostly consist of epic texts, drama is concentrated in the Classical period, and Hellenistic texts include the *Septuagint* version of the Bible.

detection without subsequent interpretation does not necessarily imply *semantic* change.

Discriminating real semantic changes from other phenomena is crucial, and at the same time not trivial for ancient languages, given that no validation with native speakers is possible. The problem of detecting real changes is entangled with the problem of the circularity of interpretation. Native speakers who want to judge the plausibility of a change in previous stages of their mother tongue have extensive extra-linguistic knowledge available, such as world- and cultural knowledge. This kind of evidence is however scarcer to modern experts of an ancient language, since most knowledge they have about the world represented in the texts derives from the ancient texts themselves. There is thus a risk of interpreting linguistic phenomena emerging from the texts with knowledge derived from them. Indeed, in the SemEval 2020 task on unsupervised LSC detection, Latin required a different treatment from modern languages, due to the non-nativeness of the experts who created the gold sense annotations (Schlechtweg et al., 2020, 4–5).

Finally, the quality and cleanness of the data can also play a role when working with corpora of ancient texts, which can be particularly affected by e.g., orthographic variation for the same word, according to regional language variety, genre, or time period, and digitization errors. For example, the systems participating in the aforementioned SemEval task generally performed worse on Latin, and the quality of the corpus is a possible reason (Schlechtweg et al., 2020, 10). This suggests that, when reusing for ancient languages methods which are effective for modern languages, the particular difficulties posed by the ancient language at hand must be taken into account.

3. Previous work

The development and progress of automatic semantic change detection follows closely the development and progress of meaning representation and processing in language technology (see Tahmasebi et al., 2021 for an overview of the early approaches). At first, count-based (co-occurrence based) methods were tested, such as Latent Semantic Analysis (Sagi et al., 2009, 2011), Positive Pointwise Mutual Information weighting of the matrices (as in Hamilton et al., 2016b and Rodda et al., 2017), or the Temporal Random Indexing (Basile et al., 2014; Caputo et al., 2015). Then, most work on LSC detection moved to using word embeddings, e.g., Kulkarni et al. (2015); Hamilton et al. (2016b,a). While such representations can capture deeper meaning relationships between words than count-based models (Baroni et al., 2014), they pose a significant problem for diachronic research: word embeddings from specific semantic spaces are not directly comparable with embeddings from any other space. To this end, various alignment strategies have been developed so that vectors from different spaces can be compared (for example, by Kim et al., 2014 and Haagsma and Nissim, 2017, the Temporal Referencing by Dubossarsky et al., 2019, TWEC by Di Carlo et al., 2019, and CADE by Bianchi et al., 2020). Most recently, contextual embeddings, such as ELMo's (Peters et al., 2018) or BERT's (Devlin et al., 2019), have been used, including many of the systems participating in the SemEval 2020 Task 1 (Schlechtweg et al., 2020, 19–22; Kutuzov and Giulianelli, 2020), and also large language models (GPT-4 in Wang and Choi, 2023), but the low-resource nature of ancient languages hampers the usability of such models (Spanopoulos, 2022). More in general, ancient languages remained at the margins of the scholarship in LSC detection, with the exception of some work on Latin (Bamman and Crane, 2011; Eger and Mehler, 2016; Perrone et al., 2021) and on Ancient Greek (Boschetti, 2009; Rodda et al., 2017; Perrone et al., 2019; McGillivray et al., 2019; Perrone et al., 2021), on which we focus next.

Building on Boschetti (2009), Boschetti (2018) used Infomap[3] to obtain two semantic spaces (BCE- and CE-texts respectively) on the TLG-E corpus, the CD-ROM version of the *Thesaurus Linguae Graecae* (Pantelia, 2001), and by analysing the neighbours of ϑάλασσα, 'sea' and ϑάνατος, 'death', observed that ϑάνατος changed its cultural and religious connotations after the rise of Christianity, differently from ϑάλασσα.

Rodda et al. (2017) also used the TLG-E corpus, again divided into BCE and CE to test changes due to the advent of Christianity, and built two semantic spaces with a count-based technique (Dinu et al., 2013). Considering words occurring more than 100 times in both subcorpora, BCE- and CE-vectors for each lemma were created so that a Pearson correlation coefficient could be computed: the lower the correlation the more the lemma was supposed to have changed its meaning. From the manual inspection of the 50 words with the lowest correlation two subsets of words emerged: a group of lemmas that came to designate key concepts of Christianity, such as παραβολή, which shifted from the meaning 'comparison' to 'parable', and a group of technical terms, which underwent a specialization or moved from a domain of knowledge to another (Rodda et al., 2017, 16–17). The work by Rodda et al. (2017) shows how Distributional Semantics can be a fruitful approach to semantic change detection in Ancient Greek, leading to the discovery of previously unknown changes thanks to the analysis of nearest neighbours. However, a

[3] http://infomap-nlp.sourceforge.net/

limitation of this work is the division into only two slices. Moreover, the CE-space is characterized by an increase in philosophical and technical works, which the authors suggest could be the explanation for alleged specializations in meaning. Indeed, as we mentioned in Section 2, the influence of genre is substantial in the limited amount of texts for different periods available for Ancient Greek.

An attempt to tackle the problem of the uneven genre distribution in the different portions of the corpus is GASC (Perrone et al., 2019), a Bayesian model that controls for genre while performing sense change detection. GASC showed an improved performance compared to the non-genre-aware SCAN model (Fremann and Lapata, 2016) and against two types of neural-embedding-based models (Perrone et al., 2019; McGillivray et al., 2019; Perrone et al., 2021). In Perrone et al. (2019) and McGillivray et al. (2019) the evaluation benchmark was a dataset of a few selected Ancient Greek words, which experts annotated by sense: μῦς, κόσμος, and ἁρμονία (Vatri et al., 2019). In Perrone et al. (2021) those words were considered as examples of 'non-changed' items, while παράδεισος and παραβολή were chosen as examples of 'changed' words. The five items were the gold standard for model evaluation.

Even if GASC performed better than other non-genre-aware models, and showed how genre information impacts word sense distribution, it was not adopted in this study. First, because it is a technique for *sense* change detection, i.e. the meaning of each word is not treated as a unity, but the different senses are kept separated. This relies on the assumption that each occurrence of a word has no sense ambiguity, but has only one sense (Perrone et al., 2019, 57–58; Perrone et al., 2021, iii). Such an assumption is problematic, especially when dealing with poetry. Moreover, to run the GASC model a parameter for the number of senses must be set, to be the same for all the words.[4] Treating word meaning as a unity, as we do here, is also an approximation, but we prefer it to forcing word usages into a preset number of senses. If such an operation can already be questionable for a modern language, for some Ancient Greek words there is not clarity and agreement even among scholars about all the senses they can have, especially across different times, genres and authors.

4. Method

By exploiting the higher-level semantic relationships that word embeddings can learn (Baroni et al., 2014), this study brings LSC detection for Ancient Greek one step further compared to previous work based on count-based models (Boschetti, 2009; Rodda et al., 2017).[5] Word embeddings are also easy to implement and not computationally intensive, and output can be extracted from them that is easy to manipulate, for example nearest neighbours to target words and cosine similarity scores. Nevertheless, using word embeddings for LSC detection brings with it the problem of space alignment (see Section 3). Methods such as the one used by Kulkarni et al. (2015) or Hamilton et al. (2016b) assume that most words did not change their meaning over time. However, such methods are problematic (Gonen et al., 2020) and have been shown to introduce noise or bias (Dubossarsky et al., 2019; Di Carlo et al., 2019).

An existing solution is CADE (Bianchi et al., 2020), first introduced as TWEC in Di Carlo et al. (2019). The advantage of TWEC/CADE is a procedure of implicit alignment, the compass method (Di Carlo et al., 2019), so that space alignment afterwards is not needed. A space (the 'compass') is first trained on the whole corpus, irrespective of slice divisions. One of the two matrices of the Word2Vec architecture (Mikolov et al., 2013) of the compass, the context (output) weight matrix, is then used to initialize the context matrices of all the slice-specific models and kept frozen. In this way, all slice-specific models are aligned because they have the same context matrix. CADE seemed suited to our study because it showed improved performance on the small corpus (50 million words) used by Bianchi et al., 2020. Even if their corpus is five times Diorisis, it was divided into 27 slices, so that the size of each individual slice should be comparable to ours (see Table 1).[6]

We adopt two measures of semantic change, the Vector Coherence (VC) and the J. They represent two different approaches to LSC detection: the VC measures the cosine similarity between the vectors of the target word in different time slices, while the J compares the nearest neighbours to the target. This second approach is close to the method developed by Gonen et al. (2020), while most previous studies measure the distance between vectors. More in detail, for each vocabulary word we calculate the VC value as the sum cosine similarities between its vectors in different time slices. The cosine similarity is calculated between all possible combination of slices, following the approach in Cassani et al. (2021). The J is based on the Jac-

[4] E.g., it was set to four in McGillivray et al. (2019).

[5] Training larger models which might yield for example contextualized embeddings was discarded due to the low-resource nature of Ancient Greek, though future work could fruitfully employ them, as long as an effective way is found to deal with the limited size of the corpus.

[6] Of course there is a difference in vocabulary size: that of the NAC-Small corpus is around 21,000 words, which occur at least 200 times in the whole corpus.

card coefficient; in particular, it is the sum of the Jaccard coefficients between all possible slice combinations. More in detail, the Jaccard coefficient for a pair of slices is measured by dividing the cardinality of the set intersection between the two lists of k nearest neighbours (i.e. the neighbours in common between the two lists) by the cardinality of the set union of the two lists (Cassani et al., 2021).

We test both measures, which were fruitfully applied to English by Cassani et al. (2021), to assess their adequacy to the case of Ancient Greek. However, in this study we do not to adopt the third measure used by Cassani et al. (2021), the Local Neighborhood Coherence (*LNC*) (Hamilton et al., 2016a, 2118). The *LNC* does not seems to be easily applicable to our corpus, since it involves measuring the cosine similarity between each word in the shared vocabulary and all the words in the union set *N* of its k nearest neighbours in all slices. We expect this to be particularly problematic, due to the scarce overlap between the nearest neighbours in the different slices. Hence we expect many words in *N* to be absent from each slice, so that the *LNC* cannot be calculated for many target-neighbour pairs.[7] Future research aiming at adapting the *LNC* to the case of Ancient Greek should carefully consider this issue. Eliminating the two smallest subcorpora (Archaic and Late Roman) could mitigate the problem.

5. Workflow, Data, and Models

Our workflow largely overlaps with that defined by Marongiu et al. (2024). First of all we identified the training data, the Diorisis Ancient Greek Corpus (Vatri and McGillivray, 2018), and we divided it into five time slices, based on the dates of the works. All words occurring less than 5 times in a slice were filtered out from that slice. An overview of the number of tokens (after frequency filtering), vocabulary size, and timespan for each slice is in Table 1. The slices roughly reflect the traditional divisions into periods of the Greek literature, but not completely (the 'Hellenistic' slice ends with the year 0), to limit unbalance in size. However, the Archaic and Late Roman slices are smaller and have a smaller vocabulary, and this contributes to limiting the shared vocabulary to 2,030 words. Future studies could expand the vocabulary by only taking into account the three, larger central slices, which share a vocabulary of 8,367 words. Eliminating the Archaic slice, in particular, would eliminate the very specific epic vocabulary of Homer and Hesiod, which represent a strong constraint to the words that can be included in the shared vocabulary. However, eliminating the first and last slice would go to the detriment of the covered timespan.

Time slice	# tokens	Vocab. size	Timespan
Archaic	229,999	3,829	beginning-500 BCE
Classical	2,628,193	14,526	499-324 BCE
Hellenistic	2,164,057	12,698	323-0 BCE
Early Roman	4,276,672	19,652	1-250 CE
Late Roman	753,907	8,578	251-500 CE

Table 1: Number of tokens, vocabulary size, and timespan per slice.

A Word2Vec language model was then trained on each of the time slices by using the CADE framework. The Continuous-Bag-of-Words architecture was trained with the following parameters: size $= 30$, siter $= 5$, diter $= 5$, workers $= 4$, sg $= 0$, ns $= 20$.[8] Subsequently, the VC and the J were calculated on the vocabulary shared among all five slices, and the words with the highest and lowest scores were examined. The metrics were also tested against known cases of semantic change.

6. Vector Coherence

The Vector Coherence of a word is the sum of its cosine similarities; to be precise, the cosine similarities between embeddings of the same word in different time slices. We calculated the cosine similarity for all possible pairs (combinations) of slices; consequently, the maximum VC value depends on the number of slices. As we have 10 possible combinations between 5 time slices, the VC thus varies between -10 and 10. A word would receive $VC = 10$ in the extreme case in which the cosine similarity was 1 in all ten slice combinations.

Results and Analysis

οὔτε, 'and not', a stop-word, is the least changed word, since it obtained the highest VC, 9.50. The 50 lemmas with the highest VC include several stop-words, such as εἰς, 'into', 9.07; οὐδέ, 'and not', 9.06; ἠέ, 'ah!', 9.01; and μήτε, 'and not', 9.00. Similarly to what found by Cassani et al. (2021, 14), other words among the 50 with the highest VC denote natural elements (πῦρ, 'fire', 9.06; χρυσός, 'gold',

[7] Cassani et al. (2021, 9) use the average word embedding of the slice to replace the embedding of the missing neighbour. It is a viable solution, but we doubt of the reliability of the *LNC* if the average vector gets used very often instead of 'real' word embeddings.

[8] More information about the possible parameters is in the source code of CADE: https://github.com/vinid/cade/blob/master/cade/cade.py.

8.93; ὕδωρ, ‘water’, 8.74; ποταμός, ‘river’, 8.72; ἥλιος, ‘sun’, 8.69), animals (βοῦς, ‘bull’, 8.71 and ἵππος’, ‘horse’, 8.62), and family relationships (παῖς, ‘child’, 8.61 and μήτηρ, ‘mother’, 8.57).[9] These results show that the VC is effective at finding stable words in our corpus of Ancient Greek.

Understanding the most changed words is more problematic, since the lowest VC scores correspond to errors in the automatic lemmatization of Diorisis. The lowest VC, 2.63, is seen for ἔσσομαι, which can be a form of εἰμί, ‘to be’, of ἕννυμι, ‘to clothe oneself’, or of ἵζω, ‘to sit’, and it should thus not be an independent lemma in the corpus.[10] Forms of different verbs have been erroneously lemmatized as ἔσσομαι in Diorisis, such as ἤχθη, an aorist passive indicative of ἄγω, ‘to lead’, or ἐξῇ, a present subjunctive of ἔξεστι, ‘it is allowed’. A similar case is the next lowest VC, 3.09, for the lemma ἄρος, ‘use, profit, help’. Most of the 2,915 wordforms of ἄρος in Diorisis are lemmatization errors, since τὸ ἄρος only occurs 14 times in the TLG corpus, of which the works in Diorisis are a subset. In particular, many wordforms lemmatized as ἄρος should have been lemmatized as ἄρα.

However, there are not only lemmatization errors among the 50 lemmas with the lowest VC. Ὀλυμπιάς, ‘Olympian’ (adjective)/’Olympic games’ (as substantive, sg. or pl.), $VC = 4.65$, is a meaningful detection. The adjective refers in the Archaic slice to the Muses (e.g., Μοῦσαι Ὀλυμπιάδες, ‘the Olympian Muses’ in Hesiod, *Theogony* 25). This clearly emerges from the ten nearest neighbours, among which we find λίγειος, ‘clear-voiced’; ἡδυεπής, ‘sweet-speaking’; παίζω, ‘dance’; μέλπω, ‘celebrate with song’. From the Classical period onwards the word can also refer to the Olympic games. The nearest neighbours in the Classical slice denote sanctuaries where important games were held or important athletes. Among the first ten we find: words related to the other three Panhellenic games (Ἰσθμοῖ, ‘on the Isthmus/at the Isthmian games’; Πυθοῖ, ‘at Pytho–Delphi–/at the Pythian games’; Νέμειος, ‘Nemean’), words related to athletes who participated in the Olympiads (e.g., Πολυδεύκης, ‘Pollux, once winner at the Olympic games’; Ἱερώνυμος, ‘Hieronymus, an Olympian athlete’), and more general terms referring to the Olympic games (e.g., Ὀλυμπίασι, ‘at the Olympic games’; Ὀλυμπιονίκη, ‘victory at Olympia’; τέθριππος, ‘with four horses yoked’). A stronger change in usage is found with the arrival of the Romans. In the Hellenistic slice the nearest neighbours are Roman names, such as Τίτος, ‘Titus’; Ἰούλιος, ‘Iulius’; Αἰμίλιος, ‘Aemilius’; and Σουλπίκιος, ‘Sulpicius’, together with the verb ὑπατεύω, ‘to be consul’. This is due to the fact that the word is used as a dating instrument,[11] (often together with the indication of the Roman consuls, for example in Diodorus Siculus, *Historical Library* 11.70). The Early Roman slice also seems to have this usage as the ten nearest neighbours are numbers (ἑκατοστός, ‘hundredth’; τριακοστός, ‘thirtieth’; ὄγδοος, ‘eighth’; etc.). In the Late Roman slice, the word occurs only six times across different authors, so that the nearest neighbours are not meaningful to characterize its usage.

Another example of a word with a low VC (3.44) which undoubtedly changed its meaning is ὕπατος, ‘highest, uppermost’. The term functions prominently as an divine epithet (especially of Zeus), besides its usage as a superlative in a spatial sense (‘highest’, ‘lowest’, ‘furthest’), a temporal sense (‘last’), or of quality (‘best’). From Roman times (though already in our ‘Hellenistic’ slice, including texts up to the year 0) the substantivized adjective refers to a consul. Examples from the ten nearest neighbours to ὕπατος in each slice show the change. In the Archaic and Classical period, these include several other divine epithets, e.g., ἀστεροπητής, ‘lightener’, epithet of Zeus; ἀγελείη, ‘driver of spoil’, epithet of Athena; Τριτογένεια, ‘Trito-born’, epithet of Athena; ἄνασσα, ‘queen’, epithet of several goddesses; ἀγυιεύς, ‘guardian of the streets’, epithet of Apollo’; παγκρατής, ‘all-powerful’, epithet of Zeus; and πολισσοῦχος, ‘protecting a city’, epithet of several guardian deities of cities. But from the Hellenistic slice the neighbours start to change, including names of politicians (e.g., Τίτος, ‘Titus’, Roman name;[12], Μάρκος, ‘Marcus’, Rom. name; Ἄππιος, ‘Appius’, Rom. name), but also words referring to political functions, such as χιλίαρχος, ‘captain/commandant’; δήμαρχος, ‘chief official of a dēmos/Roman *tribunus plebis*’; δικτάτωρ, ‘dictator’. Similarly, in the Roman slices we find δικτάτωρ, ‘dictator’; συνάρχω, ‘rule jointly’; δήμαρχος, ‘chief official of a *dēmos*/ Roman *tribunus plebis*’; ὑπατεία, ‘consulate’; ὑπατεύω, ‘to be consul’; ἀνθύπατος, ‘proconsul’; βουλή, ‘council of elders/Senate’; ἀρχαιρεσία, ‘election of magistrates’ (all in the Early Roman slice) and Αὔγουστος, ‘Augustus’; Τίτος, ‘Titus’; κράτος, ‘strength/power’; Οὐαλέριος, ‘Valerius’, Roman name; Τιβέριος, ‘Tiberius’ (all in the Late Roman slice). For this example, the nearest neighbours clearly help us see the semantic change.

Another changed word among the first 50 with the lowest VC is ἐπίσκοπος, ‘guardian, supervisor/ecclesiastical superintendent (later)’, $VC =$

[9] We use the meanings of the Ancient Greek words from the Liddell-Scott-Jones dictionary (Liddell et al., 1940) accessed through Philolog.us (March, 2005).

[10] ἔσσομαι as a headword is a heritage of the Perseus lexicon, leveraged for the automatic lemmatization of Diorisis via Diogenes (https://d.iogen.es).

[11] “An Olympiad” becomes a measure of time: the four years between the celebration of the Olympic games.

[12] Titus in this subcorpus does not refer to the Roman emperor, who lived later, but to other Roman politicians.

4.48. The change occurs most clearly in the Late Roman Period. ἐπίσκοπος is polysemous from the Archaic until the Hellenistic period, with various senses found in each time slice. In the Archaic and Classical slices, it can refer to someone having a look to obtain information, a 'spy' (e.g., Homer, *Iliad* 10.38, 10.342; Sophocles, *Oedipus at Colonus* 112), but also someone who is overseeing with the aim to protect or guard. We find both human guardians (e.g., Homer, *Iliad* 24.729; Sophocles, *Antigone* 1149) and divine guardians (e.g., Homer, *Iliad* 22.255; Aeschylus, *Seven Against Thebes* 273) as well as more abstract representations (e.g., Plato's *Laws*, 717d, where Nemesis is a guardian of Justice). There is also a sense of ἐπίσκοπος as an appointed official, an inspector or overseer with a particular task (Aristophanes, *Birds* 1021ff.; Plato, *Laws* 762d, 784a). In the Hellenistic slice, for example in the *Septuagint*, we still find these various usages of ἐπίσκοπος ('inspector, overseer': e.g., *Numbers* 4.16, *Judges* 9.28; 'divine guardian': *Wisdom of Solomon* 1.6; Callimachus, *Hymn to Artemis* 39). In the Early Roman Period, the nearest neighbours suggest similar senses, with both humans and gods as overseers. Examples of nearest neighbours are: πάρεδρος, 'sitting beside/assessor'; πρυτανεύω, 'be the president'; Ἐργάνη, 'Ergane (epithet of Athena)'; Πολιεύς, 'Polieus (epithet of Zeus)'; Κορύβας, 'Corybant (priest of Cybele in Phrygia)'. In the Late Roman Period, there is a clear change. The frequency of occurrence of the word increases drastically (322 vs 51 in the Early Roman slice), with several cases in Christian authors, 310 in Eusebius alone. The cosine similarity between the vector of ἐπίσκοπος in the Early and in the Late Roman slice is 0.32, the lowest between two consecutive slices. The sense of Christian ecclesiastic superintendent ('bishop'), already present in the New Testament, becomes prominent (e.g., Eusebius, *Ecclesiastical History* 2.1.3, *Preparatio Evangelica* 14.22.17). This sense specialization is seen in the nearest neighbours: παροικία, 'parish' in Christian texts; πρέσβυς, under which many occurrences of the comparative πρεσβύτερος were lemmatized, another Christian term for 'overseer'; Παῦλος, 'Paul'; ἀπόστολος, 'apostle'; διάκονος, 'deacon'.

Inspecting the VC scores for the items in the benchmark used for GASC (Vatri et al., 2019; Perrone et al., 2021) was only possible for κόσμος, as μῦς, ἁρμονία, παράδεισος, and παραβολή are not in the vocabulary shared between all slices. The VC assigned to κόσμος, 6.37, clashes with the assumption that the lemma did not undergo substantial semantic change (Perrone et al., 2021, viii). The cosine similarity between the vectors of κόσμος in consecutive slices and the nearest neighbours to this lemma in the five spaces explain the mismatch. The cosine similarity is very low between the Archaic and the Classical slice (0.38), while it is high for the other three combinations of consecutive slices: 0.84 between Classical and Hellenistic, 0.84 between Hellenistic and Early Roman, and 0.92 between Early and Late Roman. A low cosine similarity for all vector combinations including the Archaic causes the relatively low VC, showing how a change in usage between just two consecutive slices can strongly affect the measure. A characteristic to take into account when using the VC is thus that it is a global measure, summarizing the behaviour of a word in the whole corpus. It detects whether a change in usage happened, but doesn't say at which point(s) in time. To have insight into that, it is necessary to inspect the cosine similarity between each pair of slices. The nearest neighbours clarify the direction of the change in usage between the Archaic and Classical subcorpora. The first ten in the Archaic period are mostly related to the semantic area of chariot races (the prevailing meaning of κόσμος in this slice is 'ornament'),[13] while some nearest neighbours in the other slices point to the meanings 'world' and 'order' of κόσμος, and to the idea of management and regulation.[14] Though a change in usage is detected between the Archaic and the Classical period, we already find occurrences of κόσμος meaning 'order' in Archaic times, e.g., *Iliad* 10.472, 24.622 and *Odyssey* 8.179. The meaning 'order' was thus already in use before the Classical period (Finkelberg, 1998, 115; Elmer, 2013, 51; Horky, 2019b, 2), but, from inspection of the nearest neighbours, the meaning of 'ornament' seems to be prevalent, at least in our Archaic subcorpus. The existing literature about the meaning of κόσμος (e.g., Finkelberg, 1998, 122 and Horky, 2019a) also shows that its semantics evolved over time, most notably in the emergence of the 'cosmological' meaning 'world(-order)', although the precise timing of this, somewhere between the late Archaic and late Classical period, is debated. In conclusion, the assumption in Perrone et al. (2021, viii) should probably be rephrased as 'κόσμος did not undergo a dramatic change in usage *after the Archaic times*'.

The examined cases show how the specific characteristics of the Ancient Greek corpus constrain

[13] The ten nearest neighbours are: τέρμα, 'goal round which horses and chariots had to turn at races'; δήμιος, 'public'; νύσσα, 'turning-point in a race'; ἕδρα, 'seat, back of the horse where the rider sits'; ῥυμός, 'pole of a chariot'; δρόμος, 'race'; κόπρον, 'excrement'; πηδάλιον, 'steering-paddle'.

[14] The ten nearest neighbours in the Classical slice: οἰκοδόμημα, 'building'; ὄργανον, 'instrument, tool'; κοσμέω, 'order'; διακοσμέω, 'order/adorn'; οἴκησις, 'dwelling/administration'; ἐσθής, 'clothing'; πολεμικός, 'for/of war'; σύμπας, 'all together'; στοά, 'roofed colonnade'; κατασκευή, 'preparation/constitution'.

the LSC detection with the VC measure. In particular, lemmatization errors interfere with detection of actual semantic change, and distinguishing the two is not always straightforward. The detection of lemmatization errors proves however that the method is effective on Ancient Greek. The presence of spurious wordforms among the tokens lemmatized under a certain lemma introduces indeed noise among the contexts of that lemma, which constitute the evidence during model training. Hence the great difference between embeddings of the same lemma in different time slices, arising from inconsistency of behaviour among the wordforms. A possible correction to make the VC measure more usable, by excluding errors in lemmatization from the detections, could be to set a threshold of the VC under which the detections are discarded, i.e. when the vectors of the same lemma are *too* different between the slices. This would rely on the hypothesis that, even if a word underwent semantic change, there should still be a certain degree of consistency between its vectors in different slices.

7. J

The J of a word between two time slices, ranging from 0 to 1, is the intersection of the two lists of the top k nearest neighbours ($k = 10$ and $k = 50$ in this study) divided by the union of the two lists (thus without duplicates). The J is the sum of the Jaccard coefficients for all ten possible combinations of slices. This 'global' J score ranges between 0 and 10, depending the maximum value on the number of slices. A word would get $J = 10$ with a perfect overlap between the two lists of nearest neighbours in each combination of slices.

Results and Analysis

We first calculated J by taking into account the ten nearest neighbours ($k = 10$) in each time slice. The manual analysis of the 50 least changed words revealed that, in the same way as the VC, J is very effective at detecting stability (non-changed words). However, a difference with the VC is that among the most stable words, together with stop-words (e.g., ἐν, 'in', with the highest J, 5.48; ἐπί, 'on, upon', $J =$ 4.46; ἐκ, 'out of', $J =$; etc.), words denoting natural elements, family relationships, and animals, we find many numbers, such as δύο, 'two', $J =$ 4.23; τέσσαρες, 'four', $J =$ 3.60; and τρεῖς, 'three', $J =$ 3.40. This is due to the fact that the J doesn't take into account word vectors and cosine similarities, which are continuous values, but neighbours, i.e. categorical data. Hence to obtain a high J, exactly the same words must appear in the neighbour sets from different time slices. This is easier to achieve with numbers, which are not only stable, but also more likely to keep exactly the same neighbours (other numbers) through time. The same was observed by Cassani et al. (2021).

Word	J	VC
ἠνεμόεις	0	8.07
ἀκλεής	0	8.05
κάτος	0	3.89
ἀτερπής	0	7.28
ἰδέ	0	5.00
παρέξ	0	5.15
Ἶρις	0	6.38

Table 2: J and VC measures compared for the seven words with $J = 0$ (calculated with $k = 50$).

By increasing k to 50 (i.e. more nearest neighbours are taken into account to calculate J), more numbers appear among the 50 least changed words, obtaining the highest J. While for most other lemmas there is hardly any relatedness to the nearest neighbours after the first 10-15 neighbours, for numbers we still find other numbers and expressions of quantity much lower in the list. E.g., almost all the top 50 nearest neighbours extracted from the Hellenistic space for the target εἴκοσι, 'twenty' (the lemma with highest J, 2.97, with $k = 50$) are still clearly related to the target. Among the last ten: ὀκτακισχίλιοι, 'eight thousand'; δραχμή, 'drachma'; ἑπτακισχίλιοι, 'seven thousand'; τρεισκαίδεκα, 'thirteen'; ἐννακόσιος, 'nine hundred'. This suggests that numbers occupy a very specific part of the semantic space, mostly surrounded by other numbers or expressions of quantity. The case of numbers in Ancient Greek might confirm a potential limitation of J, which "is highest for words from the same closed, narrow semantic domains, where words are likeliest to be neighbors of each other", and "could equate diachronic coherence to closed semantic domains." (Cassani et al., 2021, 17).

On the other side of the scale, the most changed words, if J is calculated with $k = 10$ there is not enough differentiation between the most changed words, since 352 lemmas receive $J = 0$. Calculating J with $k = 50$ allows for more differentiation (even if J drops overall), so that only seven words receive $J = 0$ (Table 2). We discuss some of them.

By comparing their J score to their VC, we immediately see that two of the eight words, ἠνεμόεις, 'windy' and ἀκλεής, 'without fame', have a high VC. Their vector representation is thus stable through time. The cosine similarity between the vectors of ἠνεμόεις, in particular, remains high for all combinations of slices (min 0.75, max. 0.94). ἀκλεής is a slightly different case, with cosine similarity higher than 0.80 for most slice combinations, except for 0.77 between Archaic and Classical, 0.64 between Early and Late Roman, and 0.66 between Archaic and Late Roman. ἠνεμόεις shows that a low J, due to non-identity of neighbours across slices, can coexist with vector stability. Its nearest neighbours in the different slices exemplify this: many of them are adjectives that can be related to places or place names, but they vary across slices. Fcfor the Archaic period we find among the ten nearest neighbours: καλλιγύναιξ, 'with beautiful women'; εὔπωλος, 'abounding in horses'; Λῆμνος, 'Lemnos'; Σπάρτη, 'Sparta'; Αἰγύπτιος, 'Egyptian'; and ἐρίβωλος, 'fertile'. In the Classical period, among the first ten: Φασιανός, 'from the river Phasis', and ὑπάργυρος, 'having silver underneath'. In the Hellenistic period: Φηραί, 'Pharae (town in Messenia)'; Κραννώνιος, 'of Crannon (town in Thessaly)'; πολυτρήρων, 'abounding in doves'; Ἰνωπός, 'Inopos (river)'; etc. They are all compatible with the meaning 'windy' and do not seem to point to any semantic change for ἠνεμόεις, but, because they do not overlap across slices, ἠνεμόεις has $J = 0$.

ἀκλεής is a similar case to ἠνεμόεις. Its meaning 'inglorious' is exemplified by different related lemmas in the different slices; for example, among the ten nearest neighbours in the Classical slice there are: παραψυχή, 'consolation'; δυσκλεής, 'inglorious'; ἄταφος, 'unburied'; ἄωρος, 'untimely'; ἀνίερος, 'unholy'; ἄπολις, 'without city'. In the Hellenistic slice some of the ten nearest neighbours are: ὀιζύς, 'woe, misery'; δυστυχής, 'unlucky'; ἐξαλέω, 'to be truly recorded'; μνῆστις, 'remembrance'; αὐδή, 'voice/account'. Both lists of neighbours are related to concepts of misery and bad luck, but through different associations. The two examples suggest that a low J score does not necessarily correspond to strong semantic change.

Ἶρις is a different case, with $J = 0$ and a medium VC, 6.38. The cosine similarity is particularly low between the Archaic and the Classical period (0.48), and the analysis of the nearest neighbours shows the direction of change: Ἶρις denotes in the Archaic period the goddess Iris, messenger of the gods (among the ten nearest neighbours: ποδήνεμος, 'wind-swift, epithet of Iris'; κραιπνός, 'swift'; εὐθύς, 'straight, direct'; ἐπᾴσσω, 'rush upon'), while the–already existing[15]–meaning 'rainbow' seems to acquire more importance later (e.g., among the ten nearest neighbours in the Classical slice: ἔμφασις, 'reflection'; ἔκτασις, 'stretching out'; ἔνοπτρον, 'mirror'; ἀνάκλασις, 'reflection', together with ἀκουστής, 'listener' and ἐξακούω, 'hear'). This case suggests that a low VC paired with a low J could point to actual semantic change.

However, co-presence of extremely low VC and J values could be due to lemmatization errors. One example is κάτος, 'following', with $J = 0$ and $VC = 3.89$. An examination of some occurrences of the lemma in Diorisis (592 in total), together with its absence in the TLG, shows that wrong wordforms have been lemmatized as κάτος in Diorisis, e.g., forms of κατώτερος and Κάτων. The actual lemma κάτος seems to occur only in Plutarch's *Marcus Cato* as a translation of the Latin word *catus*.

If words with a low J do not necessarily also have a low VC, the cases of semantic change detected with the VC also received a low J: Ὀλυμπιάς has $J = 0$ (for $k = 10$, 0.09 with $k = 50$); ὕπατος: $J = 0.30$ (for $k = 10$ 10, 0.36 with $k = 50$); ἐπίσκοπος: $J = 0$ ($k = 10$, 0.06 with $k = 50$). Finally, the J score assigned to κόσμος, used as a test case for the VC measure, is 0.23 (for $k = 10$, while it is 0.40 with $k = 50$). More in detail, there is no overlap between the ten nearest neighbours in the Archaic period and in the other slices (coherently with the low cosine similarity found between the Archaic slice and any other), while there are one shared neighbour between Classical and Archaic (κατασκευή, 'state, condition'–this is the relevant meaning to the association with κόσμος) and three shared neighbours between Early and Late Roman (ἀσώματος, 'disembodied', νοερός, 'intellectual', and νοητός, 'mental'). These low values reinforce the idea of a difference in usage of the word between the Archaic and the subsequent subcorpora.

8. Conclusions

We tested two measures of semantic change, the Vector Coherence and the J, on Word2Vec word embeddings trained on a diachronic corpus of Ancient Greek. We assessed the effectiveness of both measures at detecting usage stability, and the effectiveness of VC at retrieving cases of semantic change. While the quality of the corpus impacts results when using the measures to detect change, we did find that the VC does detect actual cases of semantic change. This measure seems to be more reliable, while the J appears only useful when coupled with the VC. Lemmas with a low VC and plausible cases of semantic change also received a low J, as noticed by Cassani et al. (2021). The J measure could be biased towards words used in restricted domains instead of semantically stable words so that it should not be used alone to detect semantic change in Ancient Greek. We also observed that lemmatization errors can cause both VC and J to be extremely low.

Our analysis underscores the importance of complementing automatic detection with manual inspection, for each possible candidate to semantic change, of (i) the cosine similarities between each combination of slices and (ii) its nearest neighbours in the different slices. If a word occurs very few times in a certain subcorpus, often the nearest neighbours are not reliable indicators, and a close-reading analysis of the occurrences is the only

[15] See *Iliad* 11.27, 17.547 and Peraki-Kyriakidou (2017, 66).

method to assess whether a word underwent semantic change in a certain time frame.

Even if future work could improve the reliability of the measures, the application of computational methods has to be followed by interpretation, and the results need to be critically examined, always keeping in mind the corpus' composition. The measures adopted here cannot substitute, but only complement philological work, by suggesting unknown paths of change, or by supporting or contradicting existing theories.

9. Acknowledgements

This work was partially supported by the Young Academy Groningen through the PhD scholarship of S.S.

We also acknowledge the financial support of Anchoring Innovation. Anchoring Innovation is the Gravitation Grant research agenda of the Dutch National Research School in Classical Studies, OIKOS. It is financially supported by the Dutch ministry of Education, Culture and Science (NWO project number 024.003.012). For more information about the research programme and its results, see the website www.anchoringinnovation.nl.

We thank Federico Bianchi for the advice and fruitful discussion about the usage of the metrics.

We thank the two anonymous reviewers for their feedback.

10. Bibliographical References

David Bamman and Gregory Crane. 2011. Measuring historical word sense variation. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, pages 1–10.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247, Baltimore, Maryland. Association for Computational Linguistics.

Pierpaolo Basile, Annalina Caputo, and Giovanni Semeraro. 2014. Analysing word meaning over time by exploiting temporal random indexing. *Analysing word meaning over time by exploiting temporal Random Indexing*, pages 38–42.

Federico Bianchi, Valerio Di Carlo, Paolo Nicoli, and Matteo Palmonari. 2020. Compass-aligned distributional embeddings for studying semantic differences across corpora. *arXiv preprint arXiv:2004.06519*.

Federico Boschetti. 2009. A corpus-based approach to philological issues.

Federico Boschetti. 2018. *Copisti digitali e filologi computazionali*. CNR Edizioni. Online; accessed 03-January-2021.

Annalina Caputo, Pierpaolo Basile, and Giovanni Semeraro. 2015. Temporal random indexing: A system for analysing word meaning over time. *IJCoL. Italian Journal of Computational Linguistics*, 1(1-1):61–74.

Giovanni Cassani, Federico Bianchi, and Marco Marelli. 2021. Words with consistent diachronic usage patterns are learned earlier: A computational analysis using temporally aligned word embeddings. *Cognitive science*, 45(4):e12963.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Valerio Di Carlo, Federico Bianchi, and Matteo Palmonari. 2019. Training temporal word embeddings with a compass. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6326–6334.

Georgiana Dinu, Marco Baroni, et al. 2013. DISSECT - DIstributional SEmantics Composition Toolkit. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 31–36.

Haim Dubossarsky, Simon Hengchen, Nina Tahmasebi, and Dominik Schlechtweg. 2019. Time-out: Temporal referencing for robust modeling of lexical semantic change. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 457–470, Florence, Italy. Association for Computational Linguistics.

Steffen Eger and Alexander Mehler. 2016. On the linearity of semantic change: Investigating meaning variation via dynamic graph models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 52–58, Berlin, Germany. Association for Computational Linguistics.

David F Elmer. 2013. *The Poetics of Consent: Collective Decision Making and the Iliad*. JHU Press.

Aryeh Finkelberg. 1998. On the history of the greek koσmoσ. *Harvard studies in classical philology*, pages 103–136.

Lea Frermann and Mirella Lapata. 2016. A bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics*, 4:31–45.

Hila Gonen, Ganesh Jawahar, Djamé Seddah, and Yoav Goldberg. 2020. Simple, interpretable and stable method for detecting words with usage change across corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 538–555, Online. Association for Computational Linguistics.

Hessel Haagsma and Malvina Nissim. 2017. A critical assessment of a method for detecting diachronic meaning shifts: Lessons learnt from experiments on dutch. *Computational Linguistics in the Netherlands Journal*, 7:65–78.

William L Hamilton, Jure Leskovec, and Dan Jurafsky. 2016a. Cultural shift or linguistic drift? comparing two computational measures of semantic change. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, volume 2016, pages 2116–2121. Association for Computational Linguistics.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016b. Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1501, Berlin, Germany. Association for Computational Linguistics.

Philipp Sidney Horky. 2019a. When did kosmos become the kosmos. *Cosmos in the Ancient World*, pages 22–41.

Phillip Sidney Horky. 2019b. *Cosmos in the Ancient World*. Cambridge University Press.

Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. 2014. Temporal analysis of language through neural language models. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 61–65, Baltimore, MD, USA. Association for Computational Linguistics.

Vivek Kulkarni, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2015. Statistically significant detection of linguistic change. In *Proceedings of the 24th international conference on world wide web*, pages 625–635.

Andrey Kutuzov and Mario Giulianelli. 2020. UiO-UvA at SemEval-2020 task 1: Contextualised embeddings for lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 126–134, Barcelona (online). International Committee for Computational Linguistics.

Henry George Liddell, Robert Scott, HS Jones, and R McKenzie. 1940. *A Greek and English Lexicon A Greek and English Lexicon*. Oxford: Clarendon Press.

Jeremy March. 2005. Philolog.us.

Paola Marongiu, Barbara McGillivray, and Anas Fahad Khan. 2024. Multilingual workflows for semantic change research. *Journal of Open Humanities Data*.

Barbara McGillivray, Simon Hengchen, Viivi Lähteenoja, Marco Palma, and Alessandro Vatri. 2019. A computational approach to lexical polysemy in ancient greek. *Digital Scholarship in the Humanities*, 34(4):893–907.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Maria C Pantelia. 2001. Thesaurus linguae graecae digital library. *University of California, Irvine http://www.tlg.uci.edu*, 2:2019.

Eleni Peraki-Kyriakidou. 2017. Iris as messenger and her journey: Speech in space and time. *Time and Space in Ancient Myth, Religion and Culture, Berlin/Boston*.

Valerio Perrone, Simon Hengchen, Marco Palma, Alessandro Vatri, Jim Q Smith, and Barbara McGillivray. 2021. Lexical semantic change for ancient greek and latin. *Computational approaches to semantic change*, pages 287–310.

Valerio Perrone, Marco Palma, Simon Hengchen, Alessandro Vatri, Jim Q. Smith, and Barbara McGillivray. 2019. GASC: Genre-aware semantic change for Ancient Greek. In *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change*, pages 56–66, Florence, Italy. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Martina A Rodda, Marco SG Senaldi, and Alessandro Lenci. 2017. Panta rei: Tracking semantic change with distributional semantics in ancient greek. *IJCoL. Italian Journal of Computational Linguistics*, 3(3-1):11–24.

Eyal Sagi, Stefan Kaufmann, and Brady Clark. 2009. Semantic density analysis: Comparing word meaning across time and phonetic space. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 104–111, Athens, Greece. Association for Computational Linguistics.

Eyal Sagi, Stefan Kaufmann, and Brady Clark. 2011. Tracing semantic change with latent semantic analysis. *Current methods in historical semantics*, 73:161–183.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. Semeval-2020 task 1: Unsupervised lexical semantic change detection. *arXiv preprint arXiv:2007.11464*.

Andreas I Spanopoulos. 2022. Language models for Ancient Greek.

Nina Tahmasebi, Lars Borin, and Adam Jatowt. 2021. Survey of computational approaches to lexical semantic change detection. *Computational approaches to semantic change*, 6(1).

A Vatri, V Lähteenoja, and B McGillivray. 2019. Ancient greek semantic change-annotated datasets and code.

Alessandro Vatri and Barbara McGillivray. 2018. The Diorisis Ancient Greek Corpus. *Research Data Journal for the Humanities and Social Sciences*, 3(1):55–65.

Ruiyu Wang and Matthew Choi. 2023. Large language models on lexical semantic change detection: An evaluation. *arXiv preprint arXiv:2312.06002*.

Development of Linguistic Annotation Toolkit for Classical Armenian in SpaCy, Stanza, and UDPipe

Lilit Kharatyan, Petr Kocharov
Julius-Maximilians-Universität Würzburg
Oswald-Külpe-Weg 84, D-97074 Würzburg
{lilit.kharatyan, petr.kocharov}@uni-wuerzburg.de

Abstract

This paper presents pioneering pipelines for the UD annotation of Classical Armenian developed within the three leading linguistic annotation frameworks - UDPipe and Stanza, and SpaCy. Classical Armenian is a low resourced ancient Indo-European language, and the development of an efficient open-access NLP toolkit for it constitutes a challenging and long awaited task. The presented pipelines are trained on the Armenian Gospels with a morphological and syntactic annotation following the Universal Dependencies guidelines. The pipelines are compared to each other for the accuracy of tokenisation, POS, morphological, and syntactic parsing. Through rigorous testing, Stanza emerges as the standout system, demonstrating superior overall performance, particularly for dependency parsing and morphological tagging. In contrast, while UDPipe shows strong potential with considerable improvements in its second iteration, it does not quite reach the benchmark set by Stanza. SpaCy, despite its wide usage in NLP applications, lags behind in this comparative study, highlighting areas for potential enhancement. The paper addresses major challenges of adjusting the three frameworks to the complexities of Classical Armenian. The findings stress the importance of evaluating multiple systems in order to identify the best available solution for assisted linguistic annotation.

Keywords: Classical Armenian, automatic linguistic annotation, Universal Dependencies

1. Introduction

Classical Armenian is the earliest attested variety of the Armenian language first put to record in the early 5th century after the invention of the Armenian alphabet. The Classical Armenian literature, spanning over fifteen centuries, is of paramount importance for many aspects of culture and history, and yet it remains under-resourced regarding linguistically annotated text corpora and NLP tools. In particular, up to now, there is no open access general-purpose toolkit for the linguistic annotation of Classical Armenian. The paper presents the comparative analysis of three pipelines developed for the SpaCy, Stanza, and UDPipe frameworks and trained on a Universal Dependencies treebank of the Classical Armenian Gospels as part of the CAVaL: Classical Armenian Valency Lexicon project[1]. Although each of the models has limitations related primarily to the small size of training dataset, they constitute an important step towards developing effective tools for the assisted morphological and syntactic annotation of larger corpora of Classical Armenian texts.

1.1. Existing NLP recourses

Important resources for the lemmatization and morphological annotation of Classical Armenian have been developed within the GRE*g*ORI[2] and Calfa[3] projects. The rule-based POS-tagger and morphologizer, based on the morphological dictionary of the GRE*g*ORI project, generate annotation with multiple possible analyses for a word form in context (Kindt and Kepeklian, 2022). The analyser accurately annotates morphologically unambiguous forms found in the dictionary but the output requires disambiguation of ambiguous forms and does not annotate unregistered ones. The morphological tags rely on the unique annotation scheme of the GRE*g*ORI project (Coulie et al., 2022).

A statistical POS-tagger and lemmatizer trained on a subcorpus of the GRE*g*ORI texts (67.039 tokens) has been developed in the Calfa project (Vidal-Gorène and Kindt, 2020). The training corpus includes post-classical texts, which may depart from the Classical Armenian grammar of the 5th century and include post-classical lexical items. The following results were obtained by the Calfa team on approximately 40% of the dataset, on which the models presented in this paper were trained (see 2.1): lemmatizer f1: 94.59%; POS-tagger f1: 93.27%; morphologizer f1: 97.94% (Vidal-Gorène, p.c.). This pipeline inherits the annotation scheme of the GRE*g*ORI project and, to the best of our knowledge, is currently not in open access.

Up to now no model for the syntactic annotation

[1] https://gepris.dfg.de/gepris/projekt/518003859

[2] https://www.gregoriproject.com

[3] https://calfa.fr

of Classical Armenian has been released.

The current situation with the NLP of Classical Armenian encourages to develop a general-purpose open-access pipeline with comparable or better performance than that of the aforementioned solutions, which would provide a standardized morphological and syntactic annotation of Classical Armenian literary texts.

2. Method

2.1. Dataset

The linguistic annotation pipelines discussed in the present paper have been trained on a treebank with the morphological and syntactic annotation of the Classical Armenian translation of the Gospels (ca 82K tokens) stored in the CoNLL-U format. The treebank results from a semi-automatic conversion of the PROIEL treebank of the Classical Armenian Gospels[4] (Haug and Jøhndal, 2008) to the Universal Dependencies (UD) annotation scheme and is released as part of Universal Dependencies, v2.14.[5] Although the annotation scheme of the Classical Armenian treebank takes into account the tagsets applied to the UD treebanks in Modern Eastern and Western Armenian (Yavrumyan and Danielyan, 2020), it diverges from them in many respects insofar as significant differences of grammar require. In that regard, models trained on the Modern Armenian treebanks are not suitable for the linguistic annotation of Classical Armenian.

The treebank is split as follows: training data: 62831 tokens; development data: 8658 tokens (Matthew 4, 5; Mark 4, 5; Luke 3, 4, 5; John 3, 4, 5); test data: 10509 tokens (Matthew 6, 7, 8; Mark 6, 7; Luke 6, 7, 8; John 6, 7). The data split follows the guideline of UD, which requires that corresponding sentences in datasets that constitute a multi-lingual parallel treebank (in this case, the Gospels) end up in the same part of the dataset (training/dev/test). The split is aligned with the Ancient Greek text of the Gospels.

Training the pipelines on the UD morphosyntactic tagsets allows to obtain tools for a standardized linguistic annotation compatible with a growing number of linguistic corpora of typologically diverse languages.

2.2. Model Architecture

To effectively establish automatic linguistic annotation pipelines for Classical Armenian, a comprehensive exploration of three principal methodologies has been undertaken: UDPipe[6], Stanza[7], and SpaCy[8]. This section is dedicated to elucidating the processes involved in training, deploying, and evaluating these models. By developing several pipelines, we have pursued a practical task of empirically identifying the best solution for the assisted UD annotation of Classical Armenian.

The parameter configuration proposed for the presented models results from an approach aimed at achieving optimal performance in processing Classical Armenian data, and adhering to the standards of transparency and replicability in research. The hyper-parameters for the models were determined through a combination of empirical tuning and systematic approaches. Initially, manual tuning was performed, starting with the parameters recommended by each of the systems (Stanza, SpaCy, UDPipe). The parameters such as the number of epochs, learning rate, batch size, number of layers, units per layer, and dropout rate have been iteratively adjusted based on observed improvements in validation metrics. This process was further informed by using a random search technique, which involved sampling random combinations from the parameter space to identify promising configurations. The models have been made available online[9].

2.2.1. UDPipe

While developing annotation pipelines for Classical Armenian, UDPipe was selected as the primary tool because of its proven efficacy in processing the UD annotation stored in CoNLL-U format, including tokenization, lemmatization, POS and morphological tagging, and dependency parsing. The pipelines were trained for two versions of UDPipe, 1 (Straka and Straková, 2017) and 2 (Straka et al., 2021).

2.2.2. UDPipe 1

Tokenizer: The tokenizer component of UDPipe 1 is built around a bi-directional LSTM (Long Short-Term Memory) neural network (Sepp and Jürgen, (1997). This architecture is employed to accurately delineate both token and sentence boundaries within a given text. The operational mechanism involves the classification of each character into one of three distinct categories: ‘token boundary’, ‘sentence boundary follows’, or ‘no boundary’. Additionally, the tokenizer incorporates the *SpaceAfter=No* attribute from the MISC field of the

[4] https://github.com/proiel/proiel-treebank

[5] https://github.com/UniversalDependencies/UD_Classical_Armenian-CAVaL

[6] https://ufal.mff.cuni.cz/udpipe

[7] https://stanfordnlp.github.io/stanza/

[8] https://spacy.io/

[9] https://github.com/caval-repository/xcl_nlp

CoNLL-U dataset file. This feature allows to determine space characters and their function in the context of tokenization.

POS/Morhological Tagger and Lemmatizer: The tagger and lemmatizer for Classical Armenian are characterized by distinct but complementary functionalities. The tagger utilizes a guesser to generate various triplets of values from the fields for the universal part-of-speech tags (UPOS), language-specific part-of-speech tags (XPOS), morphological features (FEATS) in the CoNNL-U treebank, for each word. This guesser is supported by an averaged perceptron tagger, which is equipped with a fixed set of features for the disambiguation of the generated tags. Similarly, the lemmatizer operates with a guesser, producing (lemma rule, UPOS) pairs, where lemma rules are designed to modify prefixes and suffixes of a word for accurate lemma generation. This process is enhanced by considering both the suffix and prefix of words. Disambiguation in the lemmatizer, akin to the tagger, is executed by an averaged perceptron tagger.

Dependency Parser: The dependency parser integrates Parsito - a neural network-based, transition-oriented parser. Offering a suite of transition systems like the projective arc-standard, partially non-projective link2, and a fully non-projective swap system, it adeptly caters to varied syntactic structures. Key training features include embeddings for the FORM, UPOS, UFeats, and DEPREL fields of CoNNL-U. In our model, the training parameters for the aforementioned components, focusing on aspects such as guesser suffix rules and dictionary enrichment, have been meticulously configured to optimize their efficiency and precision in handling the linguistic nuances of Classical Armenian. For the UDPipe 1 model, training was specifically tailored to enhance its performance on tokenization, tagging, and parsing. Training of a tokenizer was conducted over 100 epochs at a batch size of 50, a learning rate of 0.005, and a dropout rate of 0.1. The tagging component featured two models, each configured to improve morphological analysis through guesser rules and dictionary enrichment, focusing on values of the LEMMA, XPOS, and FEAT fields. Parsing leveraged a projective transition system with embeddings for UPOS, FEAT, and FORM, executed over 40 iterations with a hidden layer of 200 and a batch size of 10. The learning rate started at 0.02, decreasing to 0.001, with L2 regularization set at 0.5 to ensure the model's generalizability.

2.2.3. UDPipe 2

Tokenizer : In the development of our UDPipe 2 pipeline, tokenization and sentence segmentation are handled using the methodology established by the baseline UDPipe 1 configuration. Specifically, a tokenizer is trained following the methodology outlined in section 2.2.2, and is integrated into the UDPipe 2 pipeline at the point of deployment. The primary distinction between the two iterations of the model resides in the adjustment of input segment size for the bi-directional GRU (Cho et al., 2014); whereas previously, the segment size was capped at 50 characters, it has been expanded to 200 characters. The selection of the optimal model is subsequently based on its performance metrics on the development dataset.

POS/Morhological Tagger and Lemmatizer: During the POS tagging phase, word embeddings undergo processing through a layered bi-directional LSTM architecture to derive contextualized embeddings. When multiple recurrent neural network (RNN) layers are employed, residual connections are implemented for layers beyond the initial one. For tag categories UPOS, XPOS and UFeats, a comprehensive dictionary is compiled, aggregating every distinct tag identified within the training corpus. However, it is important to note that our dataset does not include the XPOS field. Subsequently, a softmax classifier is employed to process these contextualized embeddings, assigning each to an appropriate class based on the pre-established tag dictionary. Given the inherent limitations of a single-layer softmax classifier, an additional dense layer equipped with tanh activation and a residual connection is introduced prior to the softmax classification stage, enhancing the model's ability to perform more complex non-linear transformations. Lemmatization is approached through their classification into specific lemma generation rules, regarded as an additional type of tag. Hence it is introduced as a fourth tag category alongside the ones mentioned above, employing a similar architectural framework for its processing.

Dependency Parser: The dependency parsing framework is predicated on a graph-based bi-affine attention parser architecture (Dozat et al., 2017). Initially, contextualized embeddings are generated by bi-directional RNNs, augmented with an artificial ROOT word at the sentence's outset. These embeddings undergo a non-linear transformation into arc-head and arc-dep representations, which are subsequently integrated through bi-affine attention to yield a distribution for each word. This distribution signifies the likelihood of every other word serving as its dependency head. An arborescence, or directed spanning tree with maximal probability, is derived utilizing the Chu-Liu/Edmonds algorithm (Chu and Liu, 1965; Jack, 1967). For the labelling of dependency arcs, a parallel process is enacted: contextualized embeddings are non-linearly mapped into rel-head and rel-dep representations and merged via bi-affine attention. This merger produces a probability distribution over potential dependency labels for each dependency edge.

The training parameters for UDPipe 2 were defined to optimize the model for the unique linguistic features of Classical Armenian. Parameters included a batch size of 32, LSTM with a cell dimension of 512 across two RNN layers, and a specific dropout rate of 0.5 to prevent overfitting. The training employed an adaptive learning rate strategy, starting at 1e-3 for the initial 40 epochs and reducing to 1e-4 for the subsequent 20 epochs, coupled with a word dropout of 0.2 to enhance generalization.

Significantly, the model has been adapted to the absence of mBERT's (Devlin et al., 2019) precomputed contextualized embeddings, which are a default expectation in UDPipe 2. This adjustment, made to accommodate the absence of support for Classical Armenian in mBERT, led to a modification in the deployment script by UDPipe developers to bypass the computation of contextualized embeddings. While this may slightly compromise accuracy, it also enhances the model's speed, presenting an advantageous trade-off. This nuanced approach to model training and deployment reflects a tailored adaptation to the challenges posed by Classical Armenian, ensuring efficient and effective linguistic processing within the constraints of available resources. Moreover, this solution paves the way for other languages lacking mBERT support, as the updated script now provides an easily replicable model for bypassing contextualized embeddings. The only requirement for others facing similar deployment constraints is to exclude the embedding from the options of the trained model.

It is important to note that, UDPipe 2, designed exclusively for Python environments and currently supported only on Linux, positions itself as a tool for research purposes rather than a direct, user-friendly successor to UDPipe 1.

2.2.4. Stanza

Tokenizer: The tokenizer employed in this pipeline exemplifies a sophisticated approach to parsing and understanding text through a unified sequence tagging model. This model, designed to accurately identify token ends, sentence boundaries, and multi-word tokens (MWTs), employs a combination of bi-directional LSTMs (BiLSTMs) and 1-D convolutional networks (CNN) for processing text at the unit level, where units are defined as single characters or syllables in accordance with language-specific orthography. This intricate setup facilitates the hierarchical classification of text segments into one of five categories: end of tokens (EOT), end of sentences (EOS), multi-word tokens (MWT), multi-word ends of sentences (MWS), and others (OTHER), through the use of binary classifiers and a gating mechanism to effectively integrate token-level information. The integration of CNNs alongside BiLSTMs aims to enhance the model's capacity for capturing local unit patterns, akin to the function of a residual connection, thereby improving the precision of the tokenizer in distinguishing between complex linguistic structures.

Lemmatizer: Stanza's lemmatizer model employs a nuanced and layered approach to the process of lemmatization, integrating both dictionary-based and neural network methodologies to address the varied and complex nature of linguistic structures it encounters. At its core, the model utilizes a dual-dictionary strategy, where the primary dictionary operates based on a combination of a word and its UPOS tag to derive lemmas, taking advantage of the predictive power of UPOS tags to enhance lemmatization accuracy while maintaining case sensitivity. In instances where the primary dictionary does not yield results, the model resorts to a secondary, word-only dictionary, providing a robust fallback mechanism. For inputs that elude the coverage of these dictionaries, the model activates its neural component, which is designed to tackle more complex lemmatization challenges that dictionaries alone cannot resolve.

This neural mechanism is intricately designed, integrating an edit classifier and a sequence-to-sequence model to handle the nuanced adjustments required for accurate lemmatization. The edit classifier is engineered to manage rare or unusually long words efficiently. It leverages the concatenated final states of an encoder, processed through a dense layer with rectified linear unit (ReLU) activation, to categorize lemmas into three distinct types: those identical to the input, those that are simply lowercased versions of the input, and those that necessitate intricate adjustments via the sequence-to-sequence model. This classification process, determined during training, allows the system to judiciously decide when to engage the more computationally intensive sequence decoder during runtime, based on the guidance from the classifier.

POS/Morhological Tagger: The POS and the morphological tagging component of the pipeline employ a sophisticated architecture centred again around a highway BiLSTM network. This network processes input that combines three distinct types of embeddings: pre-trained word embeddings, trainable frequent word embeddings for terms appearing more than seven times in the training set, and, in general applications, character-level embeddings derived from a unidirectional LSTM over each word's characters. However, for this specific implementation, character-level embeddings were not utilized, primarily for computational efficiency purposes. To compensate for the lack of character-level embeddings we implemented word2vec vectors trained on a comprehensive dataset of Classical Armenian texts[10,11] (81763 unique tokens), which significantly exceeds the pipeline training dataset. The model was trained using the Skip-gram algorithm with a context window of 5 words, a vector size of 100, and a minimum count threshold of 5, over 10 epochs.

Assigning POS tags is achieved by transforming the BiLSTM output for each word through a fully connected layer, followed by the application of an affine classifier to predict the POS tag. For XPOS tags and UFeats tags, a similar strategy is employed, but with a nuanced addition of a bi-affine classifier for XPOS, which incorporates both the state of the word's XPOS and an embedding for its UPOS tag, ensuring a harmonious relationship between the tagsets. This model is fine-tuned to minimize cross-entropy loss, aimed at capturing the diverse grammatical nuances.

Dependency Parser: The dependency parser employs a neural network architecture that integrates a highway BiLSTM to process inputs comprising pre-trained word embeddings, embeddings for frequent words and lemmas, character-level word embeddings (where available), as well as summed embeddings for XPOS/UPOS and UFeats tags. To predict unlabeled attachments, the parser utilizes a bi-affine transformation to score potential relationships between words and their heads, incorporating both edge-dependent and edge-head representations derived from the BiLSTM outputs. This method, while not explicitly accounting for the relative positions of heads and dependents, enables the model to implicitly learn such spatial relationships.

The parser also introduces mechanisms to explicitly consider the linear order and distance between words and their potential heads. By factoring in the sign and absolute difference in positions, and applying Bayes' rule under the assumption of conditional independence, it calculates the probability of a word's dependency on another, adjusting for language-specific syntactic tendencies. This calculation is further refined through deep bi-affine scorers for both linear order and distance, integrating the Cauchy distribution to model the likelihood of discrepancies in predicted arc lengths. This approach allows the parser to discourage inaccurately long or short predictions for the distance between words, enhancing its precision. This use of separate scorers for attachment and relational probabilities, alongside specialized training for each component, ensures that it not only predicts the presence of an edge but also its nature, thereby trying to achieve a more detailed and accurate parsing outcome.

For the initial iteration of the Stanza models for Classical Armenian, the training was executed using the default parameters provided by the Stanza framework[12]. This decision was made after observing that the results obtained were more than satisfactory for the scope of this project. Specifically, for tokenization, no external resources such as dictionaries were utilized, aligning with the approach to leverage innate model capabilities for linguistic processing. In contrast, word2vec vectors supplied for POS/morphological tagging were also used in dependency parsing, as required by Stanza, to enhance model performance, taking advantage of additional linguistic information embedded in these pre-trained vectors.

The dataset underwent thorough preprocessing to ensure its compatibility with the training requirements of each model component. This preparation included adjustments for multi-word tokens and sentence segmentation anomalies, tailored to the specific characteristics of Classical Armenian. The training process incorporated an early stopping mechanism to prevent overfitting, ensuring that each model component achieved optimal performance without unnecessary computational expenditure.

2.2.5. SpaCy

To explore the most effective solutions for the linguistic annotation of Classical Armenian, a decision was made to extend the pipeline investigation beyond UDpipe and Stanza, employing SpaCy for its renowned robustness and user-friendly interface.

For the SpaCy pipeline, a trainable lemmatizer, tagger, morphologizer, and parser have been selected. The models were subject to both combined and individual training and deployment. This strategy proved crucial in identifying the most effective and efficient means of processing Classical Armenian. It was observed during deployment that certain components, especially the parser and lemmatizer, demonstrated enhanced performance when operated independently. This observation underscored the necessity to take into account interactions among constituents of a pipeline with respect to the annotation tasks at hand. Additionally, for the training of the models, the word2vec vectors mentioned in the section 2.2.4 have been used.

Lemmatizer: The trainable lemmatizer of the SpaCy pipeline is intricately configured for optimal performance. It includes a Tok2Vec component for token vectorization, utilizing a MultiHashEmbed layer and a MaxoutWindowEncoder. The lemmatizer itself is structured to back off to orthographic forms, with a neural component for complex cases.

[10] https://bible.armeniancathedral.org/

[11] https://historians.armeniancathedral.org/

[12] https://github.com/stanfordnlp/stanza

Key training elements involve an Adam optimizer and a dynamic batching strategy using a compounding schedule.

POS/Morhological Tagger: Among the models tested in our SpaCy pipeline, the morphologizer and tagger demonstrated the most notable performance. The configuration of the tagger and morphologizer models has been aligned for efficient linguistic analysis. Both models utilize the *spacy.Tagger.v2* architecture to ensure consistency in their operation. They are integrated with the *spacy.Tok2VecListener.v1*, which allows them to utilize the vector representations from the shared Tok2Vec component. The Tok2Vec component averages token vectors, providing the necessary input for these models. For both the tagger and morphologizer, a label smoothing technique is incorporated to help in generalization and mitigate the risk of overfitting. Optimization for both components is managed using the Adam optimizer. Training parameters include dropout regularization and a dynamic batching strategy with a compounding schedule in order to optimize the learning process.

Dependency Parser: The parser component of the SpaCy pipeline is configured using the *spacy.TransitionBasedParser.v2* architecture. It is linked with the *spacy.Tok2VecListener.v1*, similar to other components in the pipeline, to utilize the token vectors generated by the shared Tok2Vec component. The parser model includes key parameters such as a hidden width of 128 and maxout pieces set to 3, for capturing complex syntactic relations. Similar, to the previous models, for efficient and effective training, Adam optimizer, with specified beta values and L2 regularization, has been employed. The dropout rate of 0.1 and a dynamic batching strategy, following a compounding schedule, are tailored to optimize the learning process.

3. Results

3.1. UDpipe

The UDPipe models demonstrates strong capabilities in tokenization and tagging, with especially high accuracy in identifying and classifying individual word tokens and their grammatical features. However, it encounters more challenges in multiword token recognition and sentence boundary detection, areas that could benefit from further refinement. In contrast to lemmatization, the accuracy of annotating syntactic dependencies is relatively low. The low performance of the dependency parser can be attributed to the moderate size of the dataset, which detains the model from capturing the intricacies of less frequent syntactic patterns.

The evaluation results of the trained UDPipe models 1 and 2 are presented in Table 1.

Tokenizer: The models achieve commendable results in tokenization and word segmentation. The uniformity in performance of this task across both models is attributed to the identical implementation of the tokenizer, as mentioned in the section 2.2.3. In our previous iterations, leveraging a smaller dataset (ca. 24.500 tokens) necessitated the implementation of rule-based pre-processing to augment tokenization precision and effectiveness in subsequent tasks. Remarkably, the subsequent models, trained on bigger datasets exceeding 50K tokens (including the currently best model trained on 62.831 tokens), exhibit the capability to autonomously perform this task. However, sentence segmentation still presents a challenge, evidenced by its comparatively modest results. This can indeed be linked to the peculiarities of the dataset, where the boundaries of sentences, segmented on syntactic principles, are not always formally marked. Despite these challenges, it is noteworthy that UDPipe iterations outperform other models in sentence segmentation, underscoring its relative strength in this domain.

POS-tagger, morphologizer, lemmatizer: The comparative analysis of the POS tagging, morphological tagging, and lemmatization performance between UDPipe 1 and UDPipe 2 reveals noteworthy distinctions in their efficacy. UDPipe 2 demonstrates a consistent improvement across all metrics, indicating a refined understanding and processing of grammatical features. This enhancement is particularly significant in the realm of UPOS tagging and lemmatization. The results for these two tasks are presumably superior also to those reported for the RNN pipeline of the Calfa project mentioned in Section 1.1 above. However, it is essential to note that direct comparison may not be entirely fair due to the disparity in training datasets.

The incremental advancements in UFeats tagging and the composite metric of AllTags further underscore the sophistication of UDPipe 2 in handling complex linguistic patterns. The aforementioned morphology results of the Calfa project (97.94%) appear to outperform both UDPipe iterations. Despite the observable improvements, the differences between UDPipe 2 and UDPipe 1, while statistically significant, do not overwhelmingly favour one model over the other across all tagging and lemmatization tasks. The choice between the two versions may thus hinge on specific use case requirements, computational constraints, or the need for backward compatibility. Training UDPipe 2 demands significantly more computational power and a higher level of technical skillset, making it a more resource-demanding option. Conversely, UDPipe 1 is easy to use and has a more straightforward setup process. However, for applications demanding the utmost accuracy, UDPipe 2 offers a tangible advantage.

Metric	**UDPipe 1**			**UDPipe 2**			**Stanza**
	Precision	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**	**F1 Score**
Tokens	98.82%	98.52%	98.67%	98.82%	98.52%	98.67%	**99.06%**
Sentences	90.89%	95.40%	**93.09%**	90.89%	95.40%	**93.09%**	79.51%
Words	98.83%	98.53%	98.68%	98.83%	98.53%	98.68%	**99.07%**
UPOS	96.10%	95.81%	95.95%	97.01%	96.71%	96.86%	**98.25%**
UFeats	92.69%	92.41%	92.55%	94.46%	94.18%	94.32%	**95.72%**
AllTags	91.45%	91.18%	91.31%	93.61%	93.32%	93.47%	**95.02%**
Lemmas	95.89%	95.59%	95.74%	97.26%	96.96%	97.11%	**98.67%**
UAS	82.03%	81.78%	81.90%	87.23%	86.96%	87.09%	**90.97%**
LAS	78.13%	77.89%	78.01%	83.79%	83.54%	83.66%	**87.73%**
CLAS	71.91%	71.16%	71.54%	79.74%	79.03%	79.38%	**83.20%**
MLAS	64.14%	63.48%	63.81%	72.29%	71.64%	71.96%	**80.84%**
BLEX	69.24%	68.53%	68.88%	77.79%	77.09%	77.44%	**83.20%**

Table 1: Evaluation Results of UDPipe 1, UDPipe 2, and Stanza Models

Dependency Parser: The analysis of dependency parsing results from UDPipe 1 and UDPipe 2 offers a clear illustration of the advancements made in parsing capabilities between the two versions. UDPipe 2 shows a substantial improvement across all metrics of dependency parsing, reflecting a deeper and more accurate understanding of syntactic relationships within sentences.

The parsing scores, while improved, remain lower compared to other evaluated categories. The result is clearly influenced by the limited size or diversity of the training dataset, which may not encompass the full range of syntactic constructions of Classical Armenian with sufficient frequency. However, this first open-access syntactic parser of Classical Armenian is an important step forward in the development of NLP tools for that language.

3.2. Stanza

The performance of the Stanza toolkit, as indicated by the evaluation results, is commendably strong, positioning it well against UDPipe. Stanza exhibits exceptional proficiency in handling a broad spectrum of the tasks at hand, from basic tokenization to the more complex layers of parsing and lemmatization.

Tokenizer and Lematizer: Stanza's performance in tokenization and word accuracy stands out with subtle yet notable distinctions. Unlike UDPipe 1 and 2, which maintain a consistent and high level of accuracy across tokens and words, Stanza edges forward with marginally superior token and word recognition capabilities. However, its performance in sentence segmentation significantly trails behind UDPipe (almost 13.58%), marking a distinct area for improvement.

Similar to the tokenizer, Stanza's lemmatizer significantly enhances its utility in linguistic processing. With a lemmatization accuracy of 98.67%, Stanza surpasses both versions of the UDPipe models. The synergy between Stanza's tokenization and lemmatization capabilities suggests that its advanced handling of tokens directly contributes to its exceptional performance in deriving lemmas. This interplay highlights the importance of robust tokenization as a foundation for effective lemmatization, reinforcing Stanza's superiority in addressing complex linguistic tasks.

POS/Morphological Tagger: Stanza stands out in the domain of POS and morphological tagging as well, offering superior performance in all positions. While Stanza exhibits commendable results, showing a deep understanding of linguistic nuances, its performance, although impressive, does not significantly surpass that of UDPipe 2. The gap between Stanza and UDPipe in this aspect is very narrow. While Stanza showcases slightly advanced capabilities in this task, its computational efficiency presents a consideration worth noting. Unlike UDPipe, which trains all components concurrently, Stanza adopts a sequential approach, dedicating extensive computational resources and time. Consequently, UDPipe might offer a more pragmatic choice despite its slightly lower performance metrics.

Dependency Parsing: In the realm of dependency parsing, Stanza sets a new benchmark for precision and depth in linguistic modeling. With UAS and LAS towering at 90.97% and 87.73% respectively, Stanza not only eclipses the performance of both UDPipe iterations but also significantly distances itself from SpaCy, showing its capability to discern and accurately label syntactic relationships within the text. More than the numbers, Stanza's mastery of CLAS, MLAS, and BLEX underscores its profound understanding of the complex interplay between morphological features and their syntactic functions, a testament to its advanced parsing algorithms that intricately connect contextual cues and linguistic rules.

While UDPipe provides robust baseline models,

Tokenizer & Lemmatizer		Tagger & Morphologizer		Dependency Parser	
Token Acc	97.98%	**Morphology Acc**	73.57%	**UAS**	62.80%
Token P	97.96%	**Morph micro_P**	91.07%	**LAS**	51.94%
Token R	81.84%	**Morph micro_R**	84.66%	**Sent P**	66.47%
Token F1	89.13%	**Morph micro_F1**	87.75%	**Sent R**	83.27%
Lemma Acc	92.42%	**POS Acc**	81.86%	**Sent F**	73.74%

Table 2: Evaluation Results of SpaCy Models

with its latest iteration showing commendable improvements, this stark disparity in performance between Stanza and its counterparts - particularly the sophisticated handling of complex syntactic structures and linguistic phenomena - suggests an underlying architecture that prioritizes depth of linguistic analysis over mere surface-level parsing. This comparative evaluation suggests that while Stanza demands more in terms of computational resources, its accuracy in parsing justifies this investment for cases where linguistic precision is paramount.

3.3. SpaCy

Tokenizer and Lemmatizer: In analyzing the performance of the SpaCy tokenizer and lemmatizer (Table 2) compared to that of UDPipe, both trained and tested on the same dataset, several key observations emerge.

The SpaCy tokenizer demonstrates a particular balance in precision and recall, highlighting its effectiveness in accurately identifying token boundaries while also maintaining a reasonable coverage over the entire dataset. However, when juxtaposed with tokenizers of UDPipe and Stanza, which exhibits notably higher accuracy, it becomes evident that SpaCy's tokenizer may not be as finely tuned for the specific linguistic characteristics of the dataset.

In a focused analysis of the lemmatization results, the SpaCy lemmatizer, as evidenced by its performance metrics, demonstrates a moderate level of proficiency. Given the moderate performance of this component, it is pertinent to consider the use of a lookup file for lemmatization.

It is important to note the advantage of SpaCy in offering a lookup-based lemmatizer. This approach, which relies on a pre-compiled dictionary of word forms aligned with the training dataset, is expected to yield near-optimal accuracy in lemmatization tasks. Neenless to say, the efficiency of this solution entirely depends on the quality of the dictionary.

Tagger and Morphologizer: The SpaCy morphologizer's performance, marked by a POS accuracy of 73.18% and a morphological accuracy of 75.79%, indicates a reasonable capability in identifying both POS tags and morphological features. However, a critical comparison with UDPipe 1, 2 and Stanza, which achieve higher accuracy in both UPOS and FEATS, suggests that SpaCy's model, while functional, has a lower performance in these specific areas.

The SpaCy morphologizer exhibits high precision in accurately identifying morphological features when detections are made, but its significantly lower recall suggests a challenge in consistently recognizing all pertinent morphological features in the data, leading to a precision-over-recall imbalance in its performance.

Dependency parser: The performance of the parser, as indicated by its UAS and LAS metrics, suggests a notable gap in its ability to consistently and accurately handle dependency parsing. While the parser is relatively adept at identifying syntactic dependencies, it struggles more with accurately tagging these dependencies, especially for less frequent types or types attested in complex syntactic constructions. This uneven performance across dependency types suggests that the model might benefit from more diverse training data.

4. Conclusions

The present paper evaluates three pipelines of automatic linguistic annotation developed for Classical Armenian within the UDPipe, Stanza, and SpaCy frameworks. Even though the compared models were trained on a rather limited corpus (ca. 63K tokens) they show good results and potential for further improvement by increasing the size and genre diversity of the training dataset.

The comparative study of these models demonstrates the potential for significant advancements in linguistic annotation. It highlights the critical role of dataset size, the strategic use of embeddings (or effectively bypassing them for languages with constraint training datasets lacking mBERT support in the case of UDPipe 2), and the nuanced decision-making required in selecting the most suitable framework for specific linguistic tasks. Evaluation of the results shows that the UDPipe 2 and Stanza models by far outperform the SpaCy model, and are superior or comparable to the previously developed morphological analyzer of Classical Armenian in coping with various annotation tasks. With that Stanza shows overall better performance than UDPipe 2. These results are achieved by a meticulous empirical study on training parameters for Classical Armenian, and required customization of training and deployment in order to adjust the frameworks to an under-resourced language.

5. Acknowledgments

The described annotation models have been developed as part of the "CAVaL: Classical Armenian Valency Lexicon" project funded by the Deutsche Forschungsgemeinschaft (DFG, 518003859). We thank the developers of Stanza, SpaCy and UDPipe for their indispensable help with the customization of the models, and to Chahan Vidal-Gorène for his valuable comments on the earlier version of the paper and comparative data on the efficiency of the linguistic annotator developed on the Calfa project. We acknowledge the permission of the Arak29 Charitable Foundation (`https://arak29.org`) for non-commercial use of their digital editions of Classical Armenian texts for training of the word2vec vectors used in the presented Stanza and SpaCy annotators.

6. Bibliographical References

References

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Yoeng-Jin Chu and Tseng-Hong Liu. 1965. On the shortest arborescence of a directed graph. *Scientia Sinica*, 14:1396–1400.

Bernard Coulie, Bastien Kindt, Gabriel Kepeklian, and Emmanuel Van Elverdinghe. 2022. Étiquettes morphosyntaxiques et flexionnelles pour le traitement automatique de l'arménien ancien. *Le Muséon*, 135(1-2):209–241.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Timothy Dozat, Peng Qi, and Christopher D. Manning. 2017. Stanford's graph-based neural dependency parser at the CoNLL 2017 shared task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30, Vancouver, Canada. Association for Computational Linguistics.

Dag T.T. Haug and Marius Jøhndal. 2008. Creating a parallel treebank of the old indo-european bible translations. In *Proceedings of the second workshop on language technology for cultural heritage data (LaTeCH 2008)*, pages 27–34.

Edmonds et al. Jack. 1967. Optimum branchings. *Journal of Research of the national Bureau of Standards B*, 71(4):233–240.

Bastien Kindt and Gabriel Kepeklian. 2022. Analyse automatique de l'ancien arménien. évaluation d'une méthode hybride «dictionnaire» et «réseau de neurones» sur un extrait de l'adversus haereses d'irénée de lyon. In *Proceedings of the Workshop on Processing Language Variation: Digital Armenian (DigitAm) within the 13th Language Resources and Evaluation Conference*, pages 13–20.

Hochreiter Sepp and Schmidhuber Jürgen. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Milan Straka, Jakub Náplava, and Jana Straková. 2021. Character transformations for non-autoregressive GEC tagging. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 417–422, Online. Association for Computational Linguistics.

Milan Straka and Jana Straková. 2017. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.

Chahan Vidal-Gorène and Bastien Kindt. 2020. Lemmatization and pos-tagging process by using joint learning approach. experimental results on classical armenian, old georgian, and syriac. In *Proceedings of LT4HALA 2020-1st Workshop on Language Technologies for Historical and Ancient Languages*, pages 22–27.

Marat Yavrumyan and Anna Danielyan. 2020. Universal dependencies and the armenian treebank. *Herald of the Social Sciences*, 2:231–244.

Syncing Syntax:
Building a Word Alignment Corpus through Morphological, Lemmatic, and Syntactic Annotations

Wouter Mercelis, Toon Van Hal
University of Leuven & Brepols CTLO
Blijde Inkomststraat 21, bus 3308, 3000 Leuven, Belgium, {wouter.mercelis, toon.vanhal}@kuleuven.be

Abstract

This exploratory paper discusses the potential of using automated word alignment based on manually and automatically annotated translations in a parallel corpus as a first step in extending training materials for Latin-English alignment. After describing the methodology, which is partly rule-based and partly based on machine-learning principles, the paper discusses the results of aligning the Ancient Greek and Latin New Testament as an intermediary step, using the PROIEL gold data to measure the accuracy of the approach. In a next step, this approach was extended to the Book of Judges from the Old Testament, providing high-quality aligned Ancient Greek and Latin Old Testament data. Finally, the paper describes how we plan to come to Latin-English alignments.

Keywords: Ancient Greek, Latin, Word Alignment, Treebanks; Old Testament

1. Introduction and aims

Word and sentence alignment for classical languages can serve multiple purposes. Beyond its apparent instructional and didactic use, word alignment can also facilitate tasks such as automatic word sense disambiguation (WSD) (Keersmaekers et al., 2023) or Named Entity Recognition (NER) (Hatami et al., 2021). Typically, word alignment approaches depend on a substantial corpus of training materials, which may not be readily accessible for every ancient language. This paper explores the potential of augmenting training materials by relying on both manually and automatically annotated treebanks of a parallel corpus. As to the manually annotated corpus, we can specifically rely on the New Testament treebank corpus as the backbone of the PROIEL project. Old Testament translations will offer the parallel corpus which has been (largely) automatically annotated.

This is in a very high degree an exploratory paper. It is the first step in which we want to investigate to what extent we can make use of the intensive Biblical translation activity to widen the available training material for Latin-English word alignment, which is very limited. We will examine whether our efforts might be fruitful in creating more training data in an automated way.

The structure of this paper is as follows: We begin by outlining the research context concerning the word alignment of classical languages, with a particular focus on Bible translations. We then describe the methodology we employed to leverage lemmatic, syntactic, and morphological data to achieve word-aligned versions of the Ancient Greek New Testament and its earliest translations. After presenting the results of our approach, we suggest directions for future research and potential applications.

2. Research context

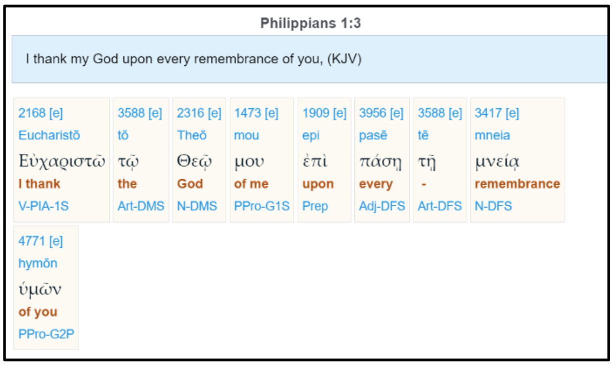
2.1 An aligned Greek-English version of the Bible

The Bible is arguably the text with the most readily available aligned translations, with the United Nations Declaration of Human Rights possibly being its closest rival in this respect. However, the majority of these translations are restricted to sentence alignment. One exception is an interlinear bible providing a sentence-aligned correspondence between the original Greek version and the King James Version (dating from the early seventeenth century)—with the English version segmented and mapped onto the Greek original (see Figure 1). For the Old Testament, no electronic interlinear translation was readily available, yet an Ancient Greek-English alignment could be established through the Apostolic Bible Polyglot, by relying on the widely used James Strong's Greek Lexicon Numbers (see Figure 2).

This mapping allows the creation of a precise word-aligned corpus between Ancient Greek and English. Consequently, if we can create reliable word-aligned translations of the Ancient Greek New Testament into other older languages, we can achieve, for example, word alignment for Latin and English (the data of which are currently very scarce), via the intermediary of Ancient Greek. Herein lies the relevance of the PROIEL project.

2.2 The PROIEL parallel corpus

The PROIEL Treebank holds a notable interest in the New Testament by providing an annotated corpus of the Ancient Greek source text, as well as its translations into Latin, Gothic, Classical Armenian, and Old Church Slavonic. These treebanks have undergone meticulous manual annotation, which includes morphological and syntactic details, and to a certain degree, information pertaining to discourse pragmatics. These annotations should be considered within the framework of a project aimed at exploring and contrasting the strategies employed in these Indo-European languages in structuring textual information. This would involve identifying and distinguishing between elements of old, new, or contrasting information to ease discourse comprehension.

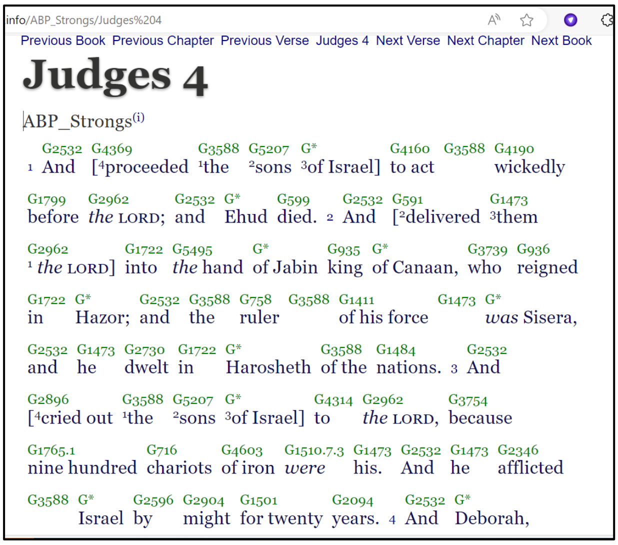


Philippians 1:3

I thank my God upon every remembrance of you, (KJV)

2168 [e] Eucharistō Εὐχαριστῶ I thank V-PIA-1S	3588 [e] tō τῷ the Art-DMS	2316 [e] Theō Θεῷ God N-DMS	1473 [e] mou μου of me PPro-G1S	1909 [e] epi ἐπὶ upon Prep	3956 [e] pasē πάσῃ every Adj-DFS	3588 [e] tē τῇ - Art-DFS	3417 [e] mneia μνείᾳ remembrance N-DFS
4771 [e] hymōn ὑμῶν of you PPro-G2P							

Figure 1: Interlinear New Testament: Ancient Greek paralleled with the King James Version (https://biblehub.com/interlinear/philippians/1-3.htm)



info/ABP_Strongs/Judges%204

Previous Book Previous Chapter Previous Verse Judges 4 Next Verse Next Chapter Next Book

Judges 4

ABP_Strongs(i)

G2532 G4369 G3588 G5207 G* G4160 G3588 G4190
1 And [^{4}proceeded ^{1}the ^{2}sons ^{3}of Israel] to act wickedly

G1799 G2962 G2532 G* G599 G2532 G591 G1473
before *the* LORD; and Ehud died. 2 And [^{2}delivered ^{3}them

G2962 G1722 G5495 G* G935 G* G3739 G936
^{1} *the* LORD] into *the* hand of Jabin king of Canaan, who reigned

G1722 G* G2532 G3588 G758 G3588 G1411 G1473 G*
in Hazor; and the ruler of his force *was* Sisera,

G2532 G1473 G2730 G1722 G* G3588 G1484 G2532
and he dwelt in Harosheth of the nations. 3 And

G2896 G3588 G5207 G* G4314 G2962 G3754
[^{4}cried out ^{1}the ^{2}sons ^{3}of Israel] to *the* LORD, because

G1765.1 G716 G4603 G1510.7.3 G1473 G2532 G1473 G2346
nine hundred chariots of iron *were* his. And he afflicted

G3588 G* G2596 G2904 G1501 G2094 G2532 G*
Israel by might for twenty years. 4 And Deborah,

Figure 2: Interlinear Old Testament: Apostolic Bible Polyglot (https://studybible.info/ABP_Strongs/Judges%204)

A relatively recently published feature is word alignment, which appears to have been implemented semi-automatically with corrections in the PROIEL Annotator web application (available through https://github.com/mlj/proiel-webapp). The algorithm works on the basis of an automatically generated bilingual dictionary, taking into account word order and morphological information too (Eckhoff, 2021). We have not made use of the application ourselves.

According to the guidelines, Greek serves as a pivotal reference point facilitating alignments between translations like the Latin New Testament and the Greek original, or the Armenian translation and the Greek original, but not directly between, e.g., Latin and Armenian translations. The potential of these alignment datasets has not yet been fully realized. In some instances, the alignments have been employed for creating visual representations (see Tauber 2020). They could potentially prove invaluable for analyzing translations and could also contribute significantly to initiatives aimed at automating word alignment in classical languages.

2.3 Computational approaches to word alignment and initiatives for the classical languages

For a comprehensive overview of recent trends in computational approaches to word alignment, one could consult Keersmaekers et al. (2023) and Li (2022), who is particularly noteworthy for providing an extensive survey with a keen focus on methodology. In the context of word alignment for ancient languages, recent contributions are from Yousef et al. (2022a; 2022b; 2023), who provide both models and datasets for various language pairs, including Ancient Greek, Latin and English. To our knowledge, attempts to operationalize parallel translations for improved word alignment in ancient languages are scarce, with Eckhoff (2021) being a prominent exception.

3. Methodology

3.1 Data preparation

After loading data files, removing duplicates and doing some data cleaning, some pre-defined mapping was applied in order to harmonize the labels for the dependency relations and the parts-of-speech columns for the Old Testament: we are relying on an automatically labelled Greek text with the Perseus AGDT conventions, while we had to make use of the labelling by LatinCy (Burns, 2023), using the conventions of Universal Dependencies for the Latin text. Proper nouns in the Greek text, which are labeled as nouns, are explicitly annotated as proper noun on the basis of capital letters in the lemma. Needless to say, such streamlining was not needed in the tests with the New Testament, as the PROIEL annotation is identical for both languages. In order to make fruitful use of the syntactic data present in the source and target language for enabling word alignment, each line of data is enriched with the PoS and syntactic relation of the head.

Both the source and the target file also contain a column with the value of the relevant sentence (e.g. "MATT 6.8"), which are used for matching the relevant sentences as well as a new column, which enumerates tokens within individual sentences.

Special attention should be paid to the articles in Greek, which are entirely absent from Latin. Hence, the articles were excluded when comparing Latin to

Greek, while they were kept when comparing Gothic to Greek.

3.2 Rule-based approach

To a large extent, our approach is rule-based. We created a script that loops through the sentences present in both corpora.

In order to keep track of the best matches with their scores for each source token, we created a "best match" list. Looping over each token in the target language for the same sentence, the script calculates a "confidence" score for each source-target pair. This score is used to evaluate how well a given token of the source language aligns with a possible target language token. No specific attention is paid to one-to-many and many-to-one mappings.

The confidence score, starting with a value of 0, is computed based on a wide range of factors. Bonus points are rewarded for each of the following correspondences between the Greek token and the translation in Latin: a match in (1) PoS (such as: verb, preposition); (2) syntactic relation (such as: subject, predicate); (3) PoS of the head; (4) syntactic relation of the head. The exact values were empirically determined, using a trial-and-error approach.

In a first run, it is not always possible to make use of the lemmatic data, except as to their relative frequency of the two corpora under comparison. In other words: when a very frequent word in Greek (e.g. καί) is matched with a very infrequent word in Latin (e.g. Barabas), this should be penalized. Hence, we compute the frequency counts of lemmas in both the source and target texts, which give rise to two dictionaries in which lemmas serve as keys and their frequencies as values. Hence, this bonus decreases as the difference in these frequencies increases. This bonus is then added to the confidence.

There are also penalties for large positional differences in token sequence. In general, the Latin translation (just like e.g. the Gothic one) largely respects the original Greek word order. While a small difference in word order can occur (e.g. due to the typically second position of a Greek particle like δέ, which cannot always be respected in the target language), it is highly improbable that a match between a Greek word at the beginning of a sentence on the one hand and a Latin word at the end of a sentence is correct. Hence, the penalization increases when the distance in index increases.

In sum, for each possible combination, a confidence score is calculated, rendering how good a match the Greek token is for the current Latin token. After considering all the Latin tokens as potential matches for the Greek token, the algorithm selects the best match as the one with the highest confidence score. We do not apply a threshold here, so our AI-model (see section 3) can clean up predictions with a low confidence score.

On the basis of the results of this first run, we can calculate lemma correspondences between the source and the target language based on the frequency and average confidence level in the aligned data. This step allows us to analyze which Greek and Latin lemmas are most commonly and reliably related to each other in the dataset based on the alignment data. After manually reviewing both the highest scores and the most frequent lemmas, we created a list of ascertained lemma matches. This list proves to be useful when running the algorithm for a second time, as from now on the more refined lemma match information, when available, can be used as a replacement of the less refined lemma frequency information.

The bonus and penalization values that are to be assigned for the different parameters are subject to discussion and trial-and-error.

Parameter	Score
Match in PoS of the token	0.4
Match in relation of the token	0.4
Match in PoS of the head	not applied in this test
Match in relation of the head	not applied in this test
Match of lemma in the lemma list	5
Frequency bonus	0.5

Table 1: Bonus values for the different parameters

Finally, the script identifies which words are present in the target dataset but missing from the alignments dataset. These are referred to as unaligned words. After iterating over each unaligned word, the script then decides what to do with these words. If the word is a form of *sum*, special rules are applied based on the Parts Of Speech (PoS) of surrounding words. If *sum* comes after a verb, it is likely that it is being used as an auxiliary verb, and therefore it is appended to a list following the previous verb. If, conversely, *sum* precedes a verb, it is inserted at the beginning of the preceding list for the following token. In either case, the confidence of the alignment for that verb is reduced.

Thus, previously unaligned target words are incorporated into the existing alignments by applying specific linguistic rules, which can be extended in a future run. Hence, this results into a more complete and coherent alignment.

3.3 Machine learning approach

In case specified confidence thresholds are not obtained (namely: 1.0), we make use of predictions stemming from a model, that has trained on manually made Greek-Latin word alignments. This model makes use of a span-extraction approach, as described in detail in Keersmaekers et al. (2023). The model mimics a Question Answering task and builds upon an existing multilingual pretrained model, PhilBERTa (Riemenschneider and Frank, 2023), which is trained on Latin, Greek and English texts. In this approach, each word in the source sentence is marked by a separation token. Consequently, the "question" takes the form of the source sentence with the designated token highlighted. The "answer" corresponds to the target token, which is located within the target sentence. The model allows all possible alignment combinations: no alignment, one-on-one alignment, one-to-many and many-to-one alignments. The predictions of the model are only used if their probability is above 0.9.

3.4 Latin-English alignment

Until this point, this paper has concentrated exclusively on the alignment between Ancient Greek and Latin. We must now pivot to consider the task of aligning Latin with English, by making use of Ancient Greek as an intermediary reference. As said, the available training data for Latin-English alignment are regrettably scarce. Consequently, our objective is to generate additional training data through automated means. In order to do this, we use the aligned Greek-English data of the Apostolic Bible Polyglot (see Figure 2), where we style one-to-many alignments as glosses, e.g. "it-came-to-pass" as one word in order to represent the corresponding single equivalent in Ancient Greek. This is crucial to maintain the link between the original source text and the pivot language text in English. Hence, we use an English-to-English alignment model called Awesome Align (Dou & Neubig, 2021) to convert our "glosses" to running text. In so doing, we possess an English-aligned textual corpus for the Ancient Greek source tokens, which, as detailed in this paper, have been previously aligned with their Latin counterparts. In doing so, we can come to an alignment of Latin with English, albeit with a number of intermediary steps. This approach can be visualized as follows (see Figure 3):

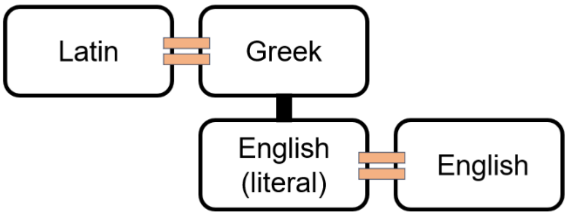


Figure 3: Schematized visualization of our Latin-English word-alignment approach

4. Results and discussion

We initially applied the described pipeline to the Greek and Latin New Testaments, which serve as reliable reference data due to their manual annotation by the PROIEL project. By excluding articles and Greek tokens without a corresponding Latin alignment, we achieved an accuracy rate with our combined approach of 91.632% (78,436 out of 85,598 source tokens), using the entire PROIEL dataset as test data, as these data were not used in training the model.

Importantly, the manual PROIEL alignment only accounts for one-to-one mappings, which is straightforward for many-to-one scenarios, as there is only a single target token predicted for each source token. However, in one-to-many mappings, several target tokens correspond with a single source token. For instance, the Greek participle ἀκούσαντες, translating to "when they hear", is represented in Latin as "cum audissent" with the same meaning. In these instances, the PROIEL gold standard indicates only "audissent" ("they hear"). A full rendering of "cum audissent" would be more precise, yet space-related errors—which likely point to one-to-many mappings—account for 43.619% of all discrepancies when compared to the gold data. It is essential to recognize that such errors do not automatically suggest our predictions are more accurate than the reference data in every case.

It was our aim to test our approach to the data of the Old Testament. For this initial approach, we limited our exploration to a qualitative investigation of the book "Judges". One notable distinction in the dataset of the Old Testament is the absence of manually annotated Greek and Latin lemmas, which complicates the lemma matching process. Nevertheless, the current state of automatic annotation for Ancient Greek and Latin lemmas yields high accuracy, which minimizes the effect of this variance. Similarly, the same principles apply to POS tags and syntactic labels.

While we have not yet managed to construct an annotated gold standard for this dataset, it appears safe to say that the trends we observed during our testing phase on the New Testament data remained consistent upon applying our approach to the Old Testament. In the end, the machine-learning method took care of alignments in 13% of the tokens.

When aligning the Old Testament, we were able to make fruitful use of the lemma data provided in the manually executed PROIEL alignment of the New Testament. This enabled us to make use of ascertained lemmas from the first round. However, there were a few misalignments in the PROIEL data, such as the incorrect alignment of οὐ and *enim* (which were aligned 8 times), which were therefore wrongly included in our list of validated lemma-based alignments. This resulted in several erroneous alignments within our data. By identifying and filtering out these inaccuracies in the PROIEL data, our methodology's effectiveness should improve. The following paragraph surveys other ways of improving the methodology explored above.

Finally, the alignment between English and Latin through the bridge of Ancient Greek and the English glosses is still in a very exploratory phase. We still need to iron out some inconsistencies, but in general, the initial results look promising. We should however acknowledge that the approach is intrinsically vulnerable, as there are multiple intermediary steps involved.

5. Avenues for further research

We aim to enhance our results for the alignment of the Old Testament data in future work through various methods. Initially, we were unable to make use of the syntactic relationships and Part-of-Speech (POS) information of the heads corresponding to the tokens being studied. The obstacle arose from the disparate syntactic annotation schemes used for Greek and Latin translations, which diverge considerably in their treatment of bridging constructions such as prepositional phrases, subordinate clauses, and coordination. This issue may be resolved by automatically adapting the syntactic trees.

Moreover, there is potential to make better use of the attributes associated with the Parts of Speech. While the grammatical cases in Latin (6 in total) do not correspond directly to those in Greek (5 in total), it is much more probable that attributes such as number, person, and degree will match between the source and target languages. Furthermore, improvement can be reached in a feasible way by refining methodologies for comparing proper names (e.g. by comparing strings) as well as by finetuning the parameters of the bonus and penalization system.

6. Acknowledgements

We gratefully acknowledge the precious suggestions by the anonymous reviewers. This work has been funded by grant no. HBC.2021.0210 of Flanders Innovation and Entrepreneurship (VLAIO) and by G052021N (Language and Ideas: Towards a New Computational and Corpus-Based Approach to Ancient Greek Semantics and the History of Ideas) of Research Foundation Flanders (FWO).

7. Bibliographical References

Burns, P. J. (2023). LatinCy: Synthetic Trained Pipelines for Latin NLP (arXiv:2305.04365). arXiv. https://doi.org/10.48550/arXiv.2305.04365.

Dou, Z.-Y., & Neubig, G. (2021). Word Alignment by Fine-tuning Embeddings on Parallel Corpora. In arXiv preprint arXiv:2101.08231.

Eckhoff, H. (2021). Automatic alignment of the Psalterium Sinaiticum and the Septuagint Psalms. *Кирило-Методиевски Студии*, *31*, 71–90.

Eckhoff, H., Bech, K., Bouma, G., Eide, K., Haug, D., Haugen, O. E., & Jøhndal, M. (2018). The PROIEL treebank family: A standard for early attestations of Indo-European languages. *Language Resources and Evaluation*, *52*(1), 29–65. https://doi.org/10.1007/s10579-017-9388-5

Hatami, A., Mitkov, R., & Corpas, G. (2021). Cross-lingual Named Entity Recognition via FastAlign: A Case Study. *Proceedings of the Translation and Interpreting Technology* Online Conference TRITON 2021, 85–92. https://doi.org/10.26615/978-954-452-071-7_010

Haug, D., & Jøhndal, M. (2008). Creating a Parallel Treebank of the Old Indo-European Bible Translations. In *Proceedings of the Second Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2008)* (pp. 27–34).

Keersmaekers, A., Mercelis, W., & Van Hal, T. (2023). Word Sense Disambiguation for Ancient Greek; Sourcing a training corpus through translation alignment. *Proceedings of the Ancient Language Processing Workshop Associated with RANLP-2023*, 148–159. https://doi.org/10.26615/978-954-452-087-8.2023_018

Li, B. (2022). *Word Alignment in the Era of Deep Learning: A Tutorial* (arXiv:2212.00138). arXiv. https://doi.org/10.48550/arXiv.2212.00138

Pedrazzini, N. (2023). *Npedrazzini/parallelbibles* [R]. https://github.com/npedrazzini/parallelbibles (Original work published 2021)

Riemenschneider, F. & Frank, A. (2023). Exploring Large Language Models for Classical Philology. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 15181-15199. https://doi.org/10.18653/v1/2023.acl-long.846*

Tauber, J. (n.d.). *Gothic-Greek Aligned New Testament*. Retrieved March 15, 2024, from https://jtauber.github.io/gothica/proiel/alignment/

Yousef, T., Palladino, C., & Shamsian, F. (2023). Classical Philology in the Time of AI: Exploring the Potential of Parallel Corpora in Ancient Language. In A. Anderson, S. Gordin, B. Li, Y. Liu, & M. C. Passarotti (Eds.), *Proceedings of the Ancient Language Processing Workshop* (pp. 179–192). INCOMA Ltd., Shoumen, Bulgaria. https://aclanthology.org/2023.alp-1.21

Yousef, T., Palladino, C., Shamsian, F., d'Orange Ferreira, A., & Ferreira dos Reis, M. (2022). An automatic model and Gold Standard for translation alignment of Ancient Greek. *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 5894–5905. https://aclanthology.org/2022.lrec-1.634

Yousef, T., Palladino, C., Shamsian, F., & Foradi, M. (2022). Translation Alignment with Ugarit. *Information*, *13*(2), Article 2. https://doi.org/10.3390/info13020065