Graphtos, a formal koinè flexion model

Léo-Nils BOISSIER, *Software Engineering student at CentraleSupélec*, Epigrapho

***Abstract*—Graphtos is a formal model specifically designed for predicting the inflection patterns of biblical Ancient Greek (koinè). With a dedicated focus on the New Testament corpus, Graphtos achieves around 98% accuracy in its predictions of word inflections. This paper presents the foundational concepts and methodologies of Graphtos, along with the notable results it has yielded. By providing valuable insights into the complexities of Ancient Greek word flexion, Graphtos enhances our understanding of the New Testament text and holds significant promise for advancements in biblical language analysis.**

CONTENTS

I. INTRODUCTION

A. Motivation and previous works

Previous methods for computational grammar often suffer from a lack of transparency, particularly concerning the phonological construction of flexed forms in languages like Ancient Greek. Addressing this limitation, Scott Fleischman's work[4] made significant strides in employing deductive and inductive approaches to unveil derivational patterns in Ancient Greek lexical forms.

B. Objective

The primary objectives of this paper are to create a computational model for Ancient Greek that achieves the following:

a) Accuracy: Develop a model that accurately predicts word inflections in biblical Ancient Greek, including both regular grammar and exceptional cases. The model aims to achieve a high level of precision in generating phonological derivations and flexed forms.

b) Explanatory Power: Provide a comprehensive and explicative analysis of the regular grammar patterns, exceptional cases, and orthographic alterations present in Ancient Greek. The model seeks to elucidate the intricate interplay between phonology and morphology, shedding light on the complexities of this ancient language.

By advancing the field of computational grammar, we aim to offer valuable insights into the phonological intricacies of Ancient Greek. It is our aspiration that this model will serve as a powerful tool for linguists, scholars, and researchers in the realms of computational linguistics and biblical language analysis, facilitating a deeper understanding of this ancient language and its linguistic structures.

C. Methodology

In this paper, we build upon Scott Fleischman's foundation and present our contribution, a novel computational model for Ancient Greek. Our approach surpasses the limitations of old methods by utilizing a base lexicon and grammar to predict ”by the book” forms (see for instances courses [3]), including those with phonetic alterations. This enables the model to generate accurate predictions based on established rules and paradigms, providing greater transparency in the analysis of Ancient Greek word inflections.

However, we acknowledge that Ancient Greek, like many languages, contains exceptional cases that deviate from regular patterns. To further enhance the accuracy of our model, we implement a minimization algorithm and manual fixes to identify and specify these exceptions. This meticulous approach not only refines the model's predictive capabilities but also ensures a more precise analysis of phonological derivations.

D. Data sources

To develop and validate our computational model for Ancient Greek, we were fortunate to have free licensing access to the following data sources. Firstly, we obtained the New Testament Greek text, along with parsing information, from the Berean Bible Society [2]. This resource furnished us with an extensive corpus of biblical Ancient Greek text, complete with valuable linguistic annotations. The parsing information played a crucial role in training and evaluating our computational model.

Secondly, we were granted access to the Greek to English Lexicon of Biblical Words from BiblicalText.com [1]. This lexicon greatly enriched our base lexicon by providing synthetic english translations for greek words.

Additionally, we would like to express our gratitude for the comprehensive course ”Grammaire grecque du nouveau testament” (Greek Grammar of the New Testament) [3]. While unpublished, this course material, generously provided by Faculté de théologie évangélique, Université Acadia, Montréal, served as a valuable reference for formalized rules and paradigms.

These diverse and generously provided data sources formed the bedrock of our research, empowering us to create a powerful computational tool capable of transparently analyzing phonological derivations and contributing to a deeper understanding of Ancient Greek linguistic structures.

E. Scope and Limitations

Our computational model for Ancient Greek exhibits both its scope and limitations, which are crucial to understanding its applicability and potential areas of improvement.

1) Scope: The model demonstrates proficiency in predicting flexed forms for Ancient Greek words present in the New Testament corpus. It efficiently utilizes deductive and inductive approaches to uncover derivational patterns, offering transparency in the analysis of word inflections. Moreover, the model successfully incorporates breathing marks to enhance phonological analysis.

We envision our model as a valuable tool for linguists, scholars, and researchers interested in computational linguistics and biblical language analysis. By providing accurate and explicative predictions for word inflections, our model contributes to a deeper understanding of the linguistic structures present in biblical Ancient Greek.

2) Limitations: While our model showcases promising results, it is important to acknowledge its limitations.

Firstly, the model does not consider the influence of contextual factors on word inflections. For instance, the last letter of a word might change when followed by a word starting with a vowel or a consonant, which is not taken into account independently.

Secondly, while breathing marks are well-supported, the model's handling of accents is limited. More complex phonological interactions are not fully captured.

Additionally, in cases where the corpus contains several variants for the same form, the model prioritizes the most regular one, possibly overlooking less common variants and resulting in fewer exceptions.

The model's focus on the New Testament corpus imposes a constraint on its generalization. As such, the model might not perform optimally on rare words or specific linguistic constructs, such as dual number, optative, or future perfect, that are infrequently encountered in the corpus.

Furthermore, given the specific scope of the New Testament, the corpus size is relatively small and may not encompass all linguistic variations and exceptional cases present in the broader context of Ancient Greek.

F. Ethical Considerations

Transparency is paramount in our research to promote academic collaboration and advance linguistic understanding.

While the model is not currently open source, we plan to explore licensing options for diverse applications, including a Greek learning application, and welcome potential partnerships to expand its reach.

Our commitment to transparency aligns with our vision of promoting openness and collaboration within the research community.

II. Model description

A. Overview

The methodology employed in this research involves building a computational model for Ancient Greek using the Dart programming language. This choice ensures the model's adaptability across various platforms, including servers, mobile devices, desktop applications, and the web.

In the first step, a database of specification objects is constructed using the `database_ingestion` package. Specifications include information for invariable words, verbs, nouns, adjectives, and pronouns, encompassing essential details like groups, endings, and tenses.

While the ”by the book” grammar can deduce specifications for many words, exceptions sometimes necessitate manual specification definition. These exceptions can be extended through code update, in the input data package `input_feed`, and an optimization script defined in the package `grammar_optimizer` enhances handling of less common exceptions automatically.

In the second step, the `grammar_model` package efficiently accesses the database of specifications to flex each word type appropriately when requested.

The model's accuracy and performance are tested using the `grammar_tester` package, which allows both single-word and bulk testing against the Bible corpus. The goal is to evaluate the model's ability to accurately predict word inflections and flexions. The model's structure is illustrated in Figure 1.

B. Low level foundation

While it is not the focus of this paper the grammar model defines a few critical component that serve as a foundation for higher level ones we will discuss in next subsections.

The first one is letter analysis. This component is able to take any greek letter, and extract the base letter and the

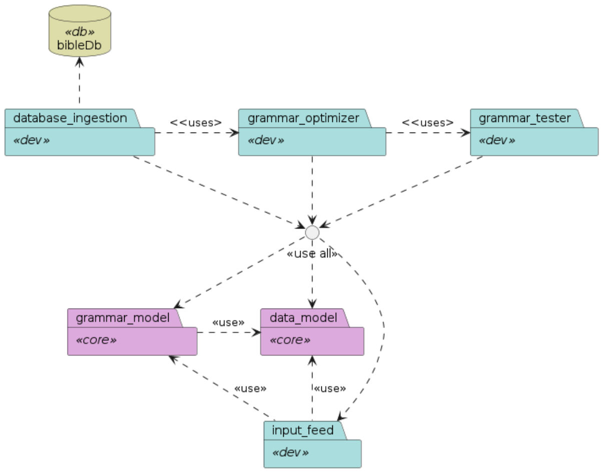


Figure 1. Overview of the methodology

modifiers (accents, breathing, iota subscript, capitalization). This serves for instance as the basis for greek text comparison (ignoring some or all of the modifiers), another critical component of the grammar model, as it is the one that allow for pattern detection, and evaluation of the model.

Another more significant component is the alteration one. In greek, some letters one next to the other might collapse into a single one, or change one the other. For instance β+σ will collapse into ψ, and ο+ντσ will change into ουσ. This component is able to detect such patterns at the boundary of two word parts, and apply the appropriate alteration.

Prefixes bring a lot of nuances to the greek words. Another component is responsible to identify them in a lexicon entry or a flexed form, to isolate them reverting appropriate alterations, or to add them back to a word later on.

These bricks are the foundation of the grammar model, and are used by all the other components.

C. Specification Objects

Specification objects are key descriptive components defined in the `data_model` package that describe the properties of various word types, enabling accurate inflection prediction. In this subsection, we present the essential classes involved, focusing on the verb example in the figure 2.

a) LexiconEntryInterface: This interface serves as a common structure for lexicon entries, applicable to different word types, including verbs, nouns, and adjectives. It encompasses fundamental attributes such as lexicalForm, definition, and type.

b) VerbLexiconEntry: As an implementation of LexiconEntryInterface, the VerbLexiconEntry class specifically handles verb entries within the lexicon. It encapsulates verb-specific information, such as the specification represented by the ConjugatorsSpec object.

c) ConjugatorsSpec: A ConjugatorsSpec is a crucial component of the verb lexicon entry, capturing the base radical of the verb (baseRadical). It is further composed of multiple TenseSpec objects, each representing the specifications for a particular tense of the verb.

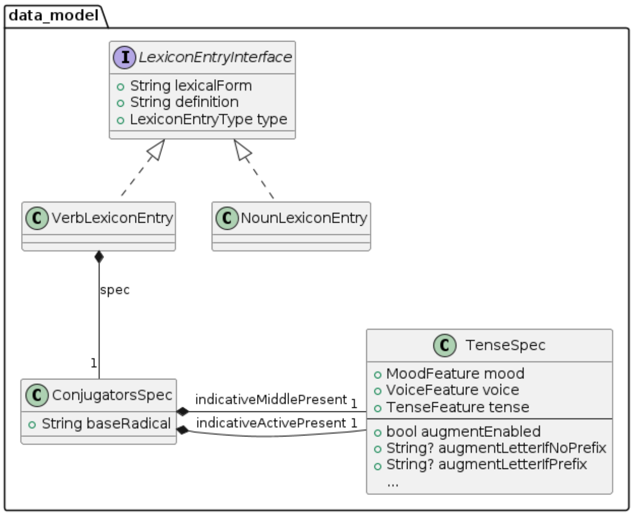


Figure 2. Simplified TenseSpec class diagram

d) TenseSpec: The `TenseSpec` class contains specifications for a specific tense of a verb. For instance, we can configure the augment behavior for the imperfect tense using properties like *augmentEnabled*, *augmentLetterIfNoPrefix*, and *augmentLetterIfPrefix*. These properties facilitate defining phonological transformations and handling exceptions, ensuring a comprehensive treatment of different verb tenses.

By utilizing these specification objects, the model can capture a very large set of behaviors. If a verb has an imperfect without augment, but only for active voice, for instance, we can deactivate only that for this verb at this tense, and then the grammar model will produce an output without augment for the specified verb. The same applies to nouns, pronouns, and adjectives, with much less features.

These specification objects are stored in a database, which is accessed by the grammar model to flex words appropriately.

D. Verb flexion

The `Conjugator` class defined in the `data_model` package is the one responsible for consuming the specification objects and flexing verbs appropriately. An optional `explanationMode` parameter can be set to true for detailed step-by-step explanations during the flexing process. The figure 3 illustrates a simplified version of the flexion process.

This process is independent of the tense being flexed, and only depends on the `TenseSpec` retrieved in the first step, so it can be tweaked as needed to take into account the worst exceptions.

E. Noun, adjective and pronoun Flexion

The `Declinator` class, defined in the `data_model` package, is responsible for flexing nouns using the specifications defined for each noun in the lexicon. Similar to the `Conjugator` class, it also has an optional `explanationMode` parameter to enable detailed step-by-step explanations during the flexion process. The flexion

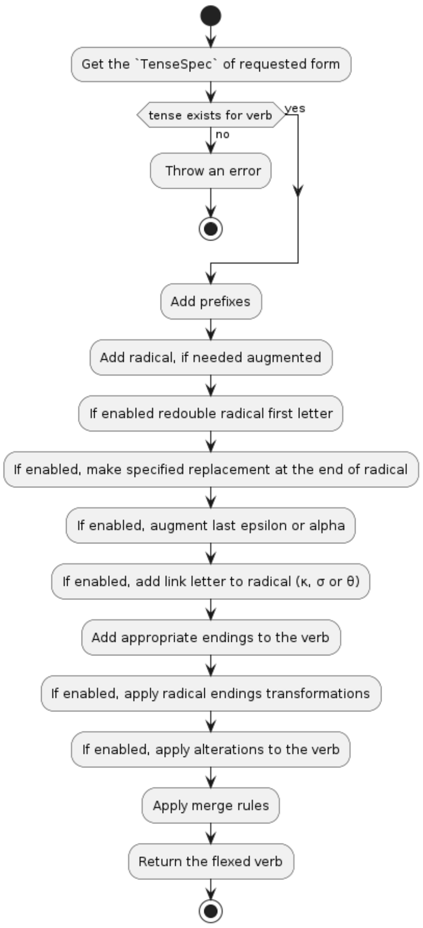

Get the `TenseSpec` of requested form
tense exists for verb
yes
no
Throw an error
Add prefixes
Add radical, if needed augmented
If enabled redouble radical first letter
If enabled, make specified replacement at the end of radical
If enabled, augment last epsilon or alpha
If enabled, add link letter to radical (κ, σ or θ)
Add appropriate endings to the verb
If enabled, apply radical endings transformations
If enabled, apply alterations to the verb
Apply merge rules
Return the flexed verb

Figure 3. Simplified prepareAlgorithm method

process for nouns is slightly different from verbs, as explained below.

The flexion of nouns involves the following steps:

1) **Add Prefixes:** The class adds any prefixes associated with the noun, which may affect the final form.
2) **Radical and Ending:** Radical and endings come from the specification object. They are retrieved and added.
3) **Apply Alterations:** Some nouns undergo alterations, such as consonant contractions. These alterations are applied to the word parts if needed.

Similar to verbs, the flexed noun is composed of several `WordPart` objects that represent the various components of the word, including prefixes, radical, and ending. Additionally, the class generate an article to accompany the noun based on the provided gender, number, and case features.

Adjectives and pronouns undergo a similar flexion process (adjectives have gender and degree to consider as well, and pronouns may have a gender too).

F. Building ConjugatorsSpec objects

The construction of `ConjugatorsSpec` objects, which encapsulate the specifications for various verb tenses, is facilitated by dedicated wrapper functions defined in the `grammar_model` package. These functions serve to select specific tense specifications based on defined criteria and apply bulk modifications to them.

To efficiently handle the basic grammar rules, the model can select large groups of tenses and apply default rules, allowing for streamlined specification handling for numerous verb forms.

However, to effectively manage exceptions and address intricate grammar cases, the model can selectively choose smaller groups of tenses or even individual tenses, allowing the application of custom rules for exceptional cases.

These wrapper functions enable the model to dynamically create `ConjugatorsSpec` objects with the appropriate specifications for each tense, based on the specified criteria and custom rules. This ensures that the model can accurately predict verb inflections and flexions, covering both regular and exceptional cases present in Ancient Greek grammar.

G. Verbs - Base verb grammar

The basic grammar is the set of rules explaining the conjugation of verbs in greek. It can be expressed as a list of selection of group of forms, and for each selection, an applied behavior. This is the most important part of the model, as it allows to flex verbs without having to specify each form individually, but does not takes into account exceptions yet. The `grammar_model` package defines how to build such a specification object from the radical only.

	Pst	Aor.	Impft	Fut	Pft	+Pdf
Active	Pst	Aor. Act.	Impft	Fut. Act.	Pft. Act.	
Middle					Pft. Mid. Pass.	
Passive		Aor. Pass.		Fut. Pass.		

Table I
TENSE GROUPS

The selection can be done at several levels. It can be a whole tense, a voice, a mood, or a common group of forms, as defined in the table I. These groups are important, because the behavior tense to be quite uniform across every forms of each of these groups.

Then come the modifications.

a) Activation: By default, all forms are enabled. (They might be disabled for exception verbs that do not have active voices for instance). Application of alterations is also enabled for all forms

b) Augment and reduplication: The table II present the behaviors activated at the begining of the verbs, either the augment or the reduplication of the first letter of the radical

	Pst	Aor.	Impft	Fut	Pft	+Pdf
Indicative		Augment			Redoub. without augment	Redoub. with augment
Infinitive			■	■		■
Participle			■	■		■
Subjunctive			■	■	■	■
Imperative			■	■	■	■

Table II
VERB SPECIFICATION AUGMENT AND REDUPLICATION

c) Link letter and lengthening: The table III present the behaviors activated at the end of the verbs, either a link letter or the lengthening of the last epsilon or alpha of the radical

	Pst	Aor.	Impft	Fut	Pft	+Pdf
Indicative		active and middle: σ passive: θ		act. midd:σ; pass:θ	Active κ, midd., pass.: augment last α, ϵ	
Infinitive			■	■		■
Participle			■	■		■
Subjunctive			■	■	■	■
Imperative			■	■	■	■

Table III
VERB SPECIFICATION LINK LETTER AND AUGMENT OF THE END OF THE RADICAL

d) The Middle and passive perfect group: At the end of radical of forms of the Middle and passive perfect tenses, the ζ is replaced with a σ, and the last vowel of radical is augmented

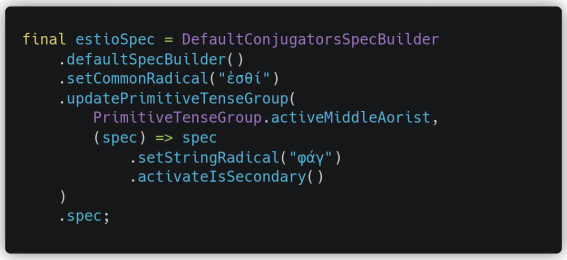
e) Endings: Default endings are set for each tense. Aorist tense also have secondary endings to specify and activate too. Perfect Indicative middle and passive have a different set of endings if the radical ends with a consonant

f) Radical ending transformations: Aorist, perfect, pluperfect and future always apply alterations between radical and endings. Imperfect too, but only for the active voice.

H. Verb exceptions, manual specification

To specify manually a verb, one can extend the default specification object just defined, by adding extra steps of selection and modification. The figure 4 illustrates the process of building a specification object for the verb ἐσϑίω (to eat), whom radical become φάγ in the aorist tense.

We start with the default configuration describe in the previous subsection, we ensure the right common radical ἐσϑίω is set for all tenses. Then we select the active and middle aorist tense group, and update all of its tense specs to have the φάγ radical and to be secondary (secondary verbs have different endings and no link letter)



```
final estioSpec = DefaultConjugatorsSpecBuilder
    .defaultSpecBuilder()
    .setCommonRadical("ἐσθί")
    .updatePrimitiveTenseGroup(
        PrimitiveTenseGroup.activeMiddleAorist,
        (spec) => spec
            .setStringRadical("φάγ")
            .activateIsSecondary()
    )
    .spec;
```

Figure 4. Example of extension of the default specification object

I. Verb exceptions, automatic optimization

Handling exceptions in verb flexion can be a challenging and error-prone task, as manually specifying each exceptional case requires significant effort and meticulous attention to detail. To address this challenge, we propose an automatic optimization method that identifies and refines the specifications for verbs that are not fully predicted by the model.

The method involves a reverse flexion algorithm, where we consider each flexed form of the verb and generate a set of tens or hundreds of possible specifications that are compatible, or nearly compatible, with the observed forms. Each specification in this set represents a combination of potential radicals, activated behaviors, and other features.

For each individual verb, we perform a search to find the single specification that best explains the largest number of flexed forms. In this process, we aim to minimize the number of errors while also minimizing the number of changes from the initial default specification.

By applying this automatic optimization method, we can effectively identify and handle exceptional cases without the need for extensive manual intervention. This approach significantly reduces the manual effort required for specifying exceptions and enhances the accuracy and performance of the model in predicting verb inflections.

In the context of our model, each use case for configuring verb inflections serves a distinct purpose. The default configuration, designed for regular verbs, forms the backbone of the system and efficiently handles a vast majority of verbs following predictable patterns. For verbs with complex exceptions and numerous occurrences, such as the verb ”εἰμί” (to be), the manual specification allows us to explicitly define the specifications for each flexed form. This approach ensures accurate predictions even when the radical cannot be deduced from the flexed forms alone.

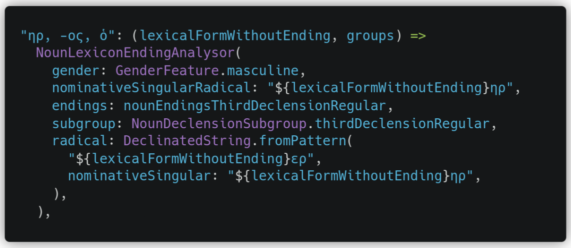
On the other hand, the automatic optimization method shines in handling a significant number of exceptions with a few tens of occurrences each. These exceptions typically deviate slightly from the regular grammar rules, making them reasonably guessable given the context of the verb's flexed forms. The automatic optimization technique swiftly identifies the most suitable specification that explains the majority of these flexed forms, effectively refining the model's predictions without the need for laborious manual intervention. This approach strikes a balance between minimizing errors and limiting deviations from the default grammar, ensuring the model's overall robustness and efficiency in handling various exceptional cases.

J. Building nouns/adjectives specifications

The specification object for nouns and adjectives typically contains endings and radical for each case, number and gender. These can often be deduced from the lexical form, and the knowledge of the possible patterns.

This is achieved by an analyzer that takes the end of the lexical form of the word, and for each possible pattern, has a definition of the radical and the endings. The analyzer then tries to match the end of the lexical form with the endings of the pattern. If it succeeds, it returns the radical and the endings. If it fails, it tries the next pattern. If no pattern matches, it returns an error.

The figure 5 illustrates one of the patterns, matching for instance ἀστήρ, -ος, ὁ (a star). For this pattern, the radical is



```
"ηρ, -ος, ὁ": (lexicalFormWithoutEnding, groups) =>
  NounLexiconEndingAnalysor(
    gender: GenderFeature.masculine,
    nominativeSingularRadical: "${lexicalFormWithoutEnding}ηρ",
    endings: nounEndingsThirdDeclensionRegular,
    subgroup: NounDeclensionSubgroup.thirdDeclensionRegular,
    radical: DeclinatedString.fromPattern(
      "${lexicalFormWithoutEnding}ερ",
      nominativeSingular: "${lexicalFormWithoutEnding}ηρ",
    ),
  ),
```

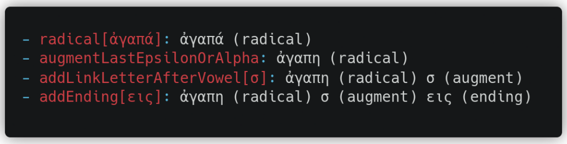
Figure 5. Example of analyzer matching the end of the lexical form of a noun with a pattern

special for the nominative singular (the epsilon is augmented), but all other cases have another radical. All (or almost all) nouns endings with this pattern behave the same.

Defining 39 such patterns for nouns and 25 for adjectives allow for a simple model explaining the flexion of most nouns and adjectives, as discussed further in the result section.

K. *Model explainability*

To achieve explainability, the model users a *FlexionStages* class to track the flexion process step-by-step. Each FlexionStage represents a specific operation performed during flexion, such as adding prefixes, applying alterations, or adding endings. It contains information about the type of operation, additional context or payload, and a list of word parts after the operation. See figure 6 for an example of the FlexionStages for the verb ”ἀγαπάω” (to love).



```
- radical[ἀγαπά]: ἀγαπά (radical)
- augmentLastEpsilonOrAlpha: ἀγαπη (radical)
- addLinkLetterAfterVowel[σ]: ἀγαπη (radical) σ (augment)
- addEnding[εις]: ἀγαπη (radical) σ (augment) εις (ending)
```

Figure 6. Example of explanation of the flexion of the verb ἀγαπάω (to love). In red, the name of the step and its arguments. Then the parts of the word just after the step is applied

The model records various types of *FlexionStages*, including AlterationFlexionStage for alterations, DoubleFirstLetterFlexionStage for doubling the first letter, and ReplacementFlexionStage for replacements at the end of radicals. By maintaining this detailed tracking, the model can provide transparent explanations of the flexion process for any input word, facilitating a better understanding of the model's decision-making and identifying exceptions or potential errors.

The main objective of the explainability feature is to facilitate a comprehensive understanding of the model's output form. This level of transparency is particularly valuable for students or users seeking to comprehend the complex rules applied to a given word. By tracking each flexion step-by-step and providing detailed FlexionStages, the model offers a complete overview of the operations performed in a single cohesive process. This enables users to grasp how individual rules are combined and applied simultaneously to handle intricate verbs, which might have been understood separately in isolation. As a result, the step-by-step tracker empowers users to comprehend the entire flexion process, ensuring a more profound insight into the model's decision-making and fostering a better grasp of complex word forms.

III. Model evaluation

A. *The test corpus*

The evaluation of the grammar model's performance was conducted through rigorous testing against the full New Testament text, which had been meticulously annotated with linguistic features such as nature, mood, voice, tense, person, degree, case, number, and gender. The extensive coverage of the New Testament text provided a comprehensive dataset for assessing the model's accuracy across various linguistic dimensions. The assessment process involved flexing each encountered word using the model and subsequently comparing the generated forms against the annotated forms present in the New Testament text.

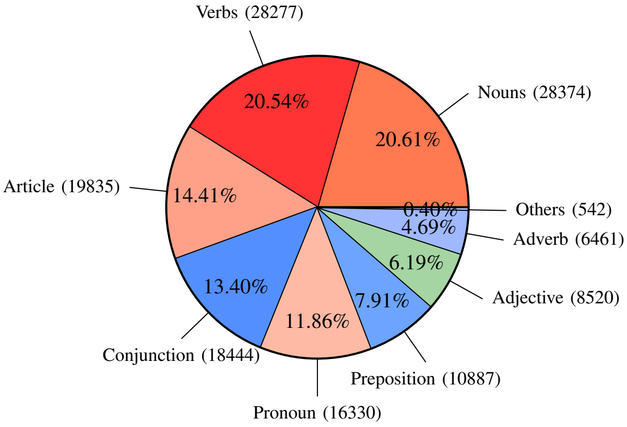


Figure 7. Frequencies of the most common natures in the dataset. In orange the noun group, in red the verb group, in blue the invariable words, and in green the adjectives

The analysis of the corpus shows that concentrating efforts on certain linguistic natures within the Greek grammar model is of paramount importance due to their prevalence and inherent characteristics. The frequencies of different linguistic natures within a dataset of 137,670 occurrences are indicative of the distribution and significance of each nature. Nouns (20.61%) and verbs (20.54%) emerge as the most frequent linguistic categories, underscoring their pivotal role in the Greek language. These two categories warrant focused attention given their high occurrence rates and the essential role they play in constructing meaningful sentences.

Furthermore, it is essential to prioritize the implementation of articles (14.41%) and invariable words, despite their relatively lower frequencies, due to the straightforward nature of their implementation. While articles and invariable words may appear less frequently compared to nouns and verbs, their inherent simplicity in terms of grammar and morphology makes them prime candidates for accurate implementation. Ensuring precision in handling these linguistic elements can serve as foundational stepping stones towards enhancing the overall accuracy and functionality of the grammar model.

This emphasis on key linguistic natures aligns with an effective strategy to optimize the grammar model's performance, focusing efforts on both high-frequency components and those that offer relatively low-hanging fruit in terms of implementation complexity. Such a strategic approach aims to strike a balance between addressing the most common linguistic constructs while also capitalizing on opportunities for accurate implementation and refinement in specific linguistic domains. To provide a comprehensive understanding of the significance of this approach, Table I presents a breakdown of the frequencies of each nature within the evaluation dataset.

B. Evaluation methodology

To ensure a comprehensive evaluation, occurrences of errors were tallied by considering the frequency of word appearances in the text. In this approach, if the same error manifested in a word that occurred multiple times, each instance of the error was individually recorded, effectively assigning greater significance to more frequently occurring words.

The evaluation procedure adopted a binary criterion for assessing the correctness of the model's output. Specifically, for each encountered word form, the model-generated flexed form was compared against the corresponding forms specified in the New Testament text for the relevant linguistic features. If the model-produced form matched at least one correct variant found in the text (disregarding accent accuracy, which is a current limitation of the model), it was deemed correct. The evaluation further distinguished between two categories of errors: instances where the model failed to produce a required form (termed ”not implemented”) and instances where the model confidently generated an incorrect form.

By adopting this comprehensive testing methodology, the evaluation process facilitated a neutral and systematic assessment of the grammar model's performance across the diverse array of linguistic features present in the New Testament text. The methodology provided a robust framework for quantifying the model's accuracy and identifying areas of strength and improvement, contributing to a thorough understanding of its capabilities and limitations.

C. By the book grammar

The analysis of the first model's results reveals distinct patterns of accuracy across different linguistic natures, shedding light on the model's performance and areas for improvement. To facilitate a comprehensive understanding, we will examine the accuracy of the grammatical model across three distinct groups: (nouns and adjectives), verbs, and (articles, conjunctions, prepositions, particles, and interjections). The results are presented in figure 8.

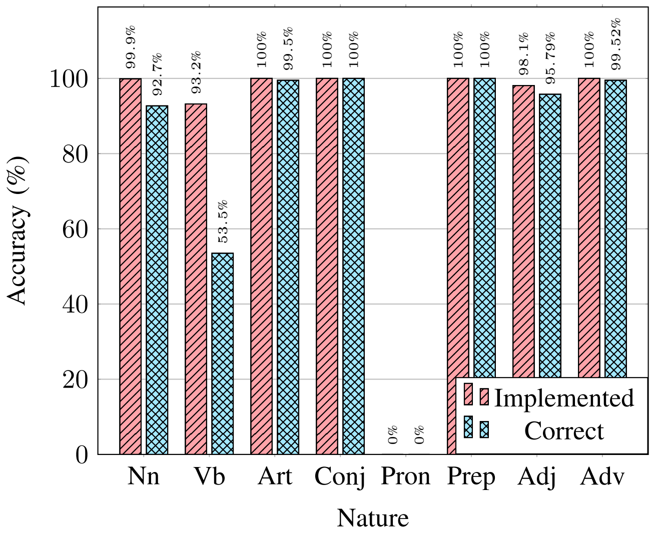


Figure 8. Accuracy of the basic model, with no exception specified.

a) Nouns and Adjectives: The accuracy for nouns and adjectives is in both case above 90% without considering any exception. This result could be attributed to the relatively well-defined and predictable patterns within these linguistic categories. Both nouns and adjectives typically exhibit consistent morphological changes based on case, number, and gender, which allows the model to flexibly handle variations with a higher degree of accuracy. Even if the model has to consider a lot of different patterns, they are easily identifiable to each lexical form.

b) Verbs: In stark contrast, the performance of verbs presents a notable challenge with a base accuracy of 53.49%. The intricate conjugation patterns, involving mood, voice, tense, person, and number, contribute to the complexity of verbs. The model's struggle to accurately capture these intricate patterns leads to a lower accuracy rate. Additionally, the highly irregular nature of certain verb forms further exacerbates the challenges faced by the model in this domain.

c) Articles, Conjunctions, Prepositions, Particles, Interjections: The collective accuracy for articles, conjunctions, prepositions, particles, and interjections is relatively high, above 99%. These linguistic elements often have more straightforward and invariable forms, allowing the model to accurately predict their flexed forms with a high degree of success. The simplicity of these elements' grammar and morphology contributes to their overall accurate implementation within the model.

In summary, the initial model's accuracy across different linguistic natures highlights the varying degrees of complexity inherent in Greek grammar. While nouns and adjectives demonstrate a strong performance owing to their well-defined patterns, the lower accuracy of verbs underscores the intricate and irregular nature of their conjugations. On the other hand, the high accuracy achieved with articles, conjunctions, prepositions, particles, and interjections reflects the relatively straightforward grammar rules governing these elements. These insights emphasize the need for focused attention on refining the treatment of verbs to enhance the overall accuracy and performance of the grammatical model.

D. Manually curated grammar

The outcomes following the curation of the most prevalent exceptions within the New Testament data reveal a notable enhancement in the grammatical model's accuracy with limited efforts. These results can be evaluated within the context of distinct linguistic categories, illustrating the impact of exception curation on each nature. The results of the updated model are presented in figure 9.

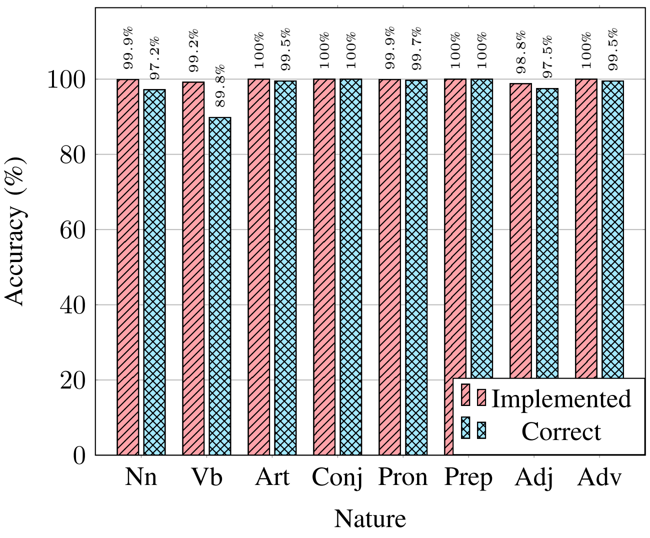


Figure 9. Model with manually specified exceptions.

a) Nouns and Adjectives: : The accuracy for nouns and adjectives has seen a substantial increase to 97.19% and 97.45%, (+4.5pt% and +17pt% respectively). To achieve such an improvement only 12 nouns and 15 adjectives were curated, for instance Ἰησοῦς, -ου, ὁ (Jésus, 916 occurrences) or μέγας (large, 243 occurrences).

b) Verbs: : While the accuracy of verbs has seen an improvement to 89.78%, it still remains relatively lower compared to other categories. The challenges posed by the conjugation patterns and irregularities in verbs continue to influence the model's performance. Nevertheless, the curation of exceptions has resulted in a significant enhancement, highlighting the impact of targeted improvements in addressing specific linguistic complexities. To achive this +36.3pt%, 72 base verbs have been manually specified. As most of them come with a few variants with different prefixes, this curation represents a total of 338 unique verbs.

c) Pronouns: : There is little to no regularity in the declension of pronouns, but they are so few different ones that they all ended being manually curated. Thats why pronouns have a 0 implementation and accuracy score in the previous model. Manual curation led to 99.7% accuracy.

The curation of exceptions within the New Testament data has led to substantial accuracy improvements in most linguistic categories. While nouns, adjectives, and certain irregular verbs have experienced notable enhancements, the challenges posed by the intricate conjugation patterns of verbs persist.

E. Final model

The subsequent phase of automatic optimization has yielded significant additional improvements to the grammatical model's performance. Specifically, in the domain of verbs, the optimization process has led to a reduction in incorrect occurrences from 2672 to 1009, resulting in an increased accuracy rate from 89.78% to 95.66%. This notable enhancement underscores the effectiveness of the optimization algorithms in addressing the intricacies and irregularities inherent to verb conjugation patterns. Furthermore, the aggregate impact of automatic optimization is evident in the overall results, as the total number of incorrect occurrences has been substantially reduced from 3718 to 2055, elevating the accuracy rate from 96.97% to 98.18%. These outcomes underscore the role of algorithmic refinement in refining the model's handling of various linguistic nuances and patterns, ultimately contributing to a more robust and accurate grammatical analysis.

The figure 10 presents the results of the final model. The scale is updated around 95% and 100% to better show the differences between the different categories.

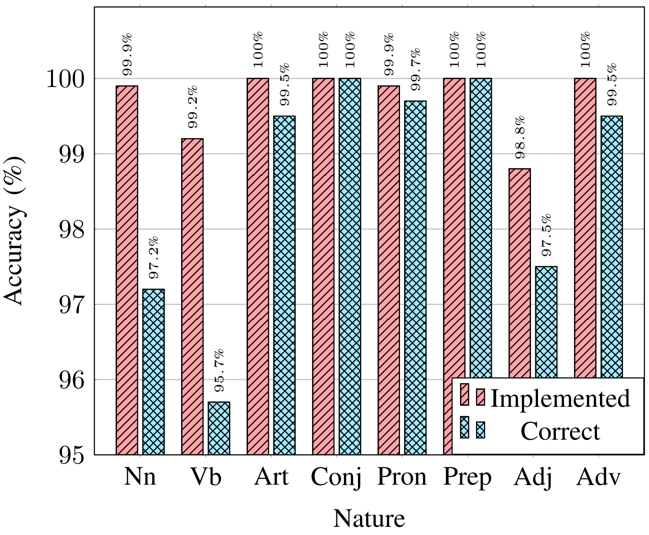


Figure 10. Final model results after automatic optimization of verbs. To improve readability, the scale of the y-axis is increased to 95-100%

IV. Model presentation

A. Frontend application

In this section, we unveil the web application interface that provides a direct and accessible gateway to our grammar model. Through this frontend application, users can effortlessly navigate the model's flexion capabilities, delve into parameter settings, and witness the intricate interplay between linguistic input and output predictions.

Our frontend interface bridges the gap between advanced linguistic models and intuitive user interaction. By inputting text and specifying grammatical categories users can witness the model's capabilities in generating accurate inflections. In the following subsections, we will explore the various features of the frontend application, highlighting the model's performance and capabilities.

B. Form generation

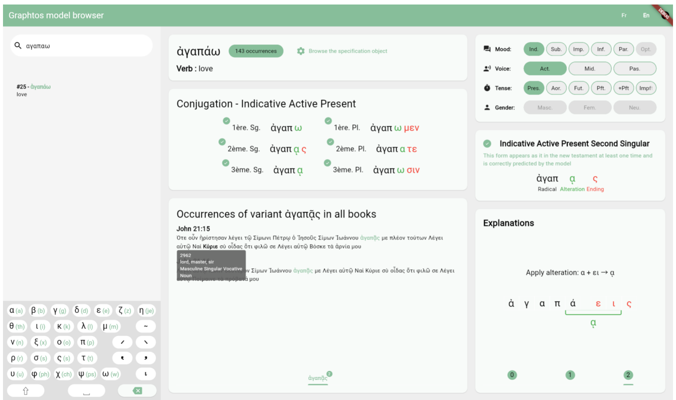
The frontend application allows users to search for greek text using a greek keyboard, a mapping to latin keyboard, or a strong number. Once selected, Three area are displayed

- An overview are, with the lexical form and a definition of the word
- A form selector area, for the user to select a mood/voice/tense if it is a verb, a degree if it is an adjective, ...
- A flexion table for the selected features

When the user select a single form in the flexion table, three additional areas are displayed:

- A list of verses containing the selected form in the new testament.
- A parsing of the parts that compose the flexed word
- A step by step explanation of the flexion process

The figure 11 shows the exploration of verb ἀγαπάω (love) in the indicative mood, active voice, and present tense. The user then selected the second singular form. The occurrences area highlights the two occurrences of the form in the new testament, whereas the explanation area explain for instance how the alteration α+ει led to the form ἀγαπᾳς.


Graphtos model browser
Fr
En
DEBUG
αγαπαω
#25 - ἀγαπάω
love
ἀγαπάω
143 occurrences
Browse the specification object
Mood: Ind. Sub. Imp. Inf. Par. Opt.
Voice: Act. Mid. Pas.
Tense: Pres. Aor. Fut. Pft. +Pft Impf.
Gender: Masc. Fem. Neu.
α (a) β (b) γ (g) δ (d) ε (e) ζ (z) η (je)
θ (th) ι (i) κ (k) λ (l) μ (m) ~
ν (n) ξ (x) ο (o) π (p)
ρ (r) σ (s) ς (s) τ (t)
υ (u) φ (ph) χ (ch) ψ (ps) ω (w)
ἀγαπᾷς 2
0
1
2

Verb : love

Conjugation - Indicative Active Present

1ère. Sg.	ἀγαπ ω	1ère. Pl.	ἀγαπ ω μεν
2ème. Sg.	ἀγαπ ᾳ ς	2ème. Pl.	ἀγαπ α τε
3ème. Sg.	ἀγαπ ᾳ	3ème. Pl.	ἀγαπ ω σιν

Indicative Active Present Second Singular

This form appears as it in the new testament at least one time and is correctly predicted by the model

ἀγαπ ᾳ ς

Radical Alteration Ending

Occurrences of variant ἀγαπᾷς in all books

John 21:15

Ότε οὖν ἠρίστησαν λέγει τῷ Σίμωνι Πέτρῳ ὁ Ἰησοῦς Σίμων Ἰωάννου ἀγαπᾷς με πλέον τούτων Λέγει αὐτῷ Ναί **Κύριε** σὺ οἶδας ὅτι φιλῶ σε Λέγει αὐτῷ Βόσκε τὰ ἀρνία μου

2962
lord, master, sir
Masculine Singular Vocative
Noun

...ν Σίμων Ἰωάννου ἀγαπᾷς με Λέγει αὐτῷ Ναί Κύριε σὺ οἶδας ὅτι φιλῶ σε Λέγει ... μου

Explanations

Apply alteration: α + ει → ᾳ

ἀ γ α π ά ε ι ς

ᾳ

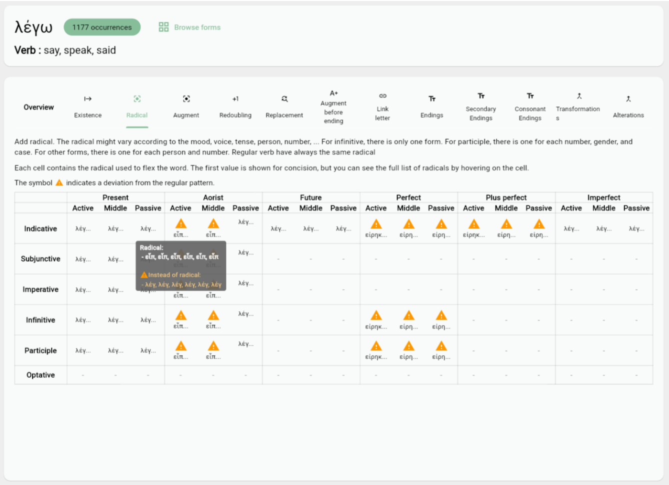
Figure 11. Grammar model

C. Model parameters exploration

The user can also explore the model parameters. The figure 12 shows the exploration of the verb λεγω (say). For a given word, the specification objects described in subsection II-C are displayed as tables. For verbs, we also compute the specification again ignoring the exception definitions. This way we can display both the actual specification and the specification that would be used if no exception was defined. This allows the user to see the impact of the exceptions on the specification.

For instance, for λεγω, the aorist is replaced with the word εἶπον that hold the past value of λεγω. The figure 12 shows the specification of the verb λεγω (say). The orange marks highlight that an exception was applied for all the moods, for aorist active and middle, perfect and pluperfect.

When such an exception occurs, because of the minimization process, it tends to try to occupy entire columns or rows, so we reduce the risk of over-fitting to the few forms we know from the new testament.



λέγω

Verb : say, speak, said

Add radical. The radical might vary according to the mood, voice, tense, person, number, ... For infinitive, there is only one form. For participle, there is one for each number, gender, and case. For other forms, there is one for each person and number. Regular verb have always the same radical

Each cell contains the radical used to flex the word. The first value is shown for concision, but you can see the full list of radicals by hovering on the cell.

The symbol ⚠ indicates a deviation from the regular pattern.

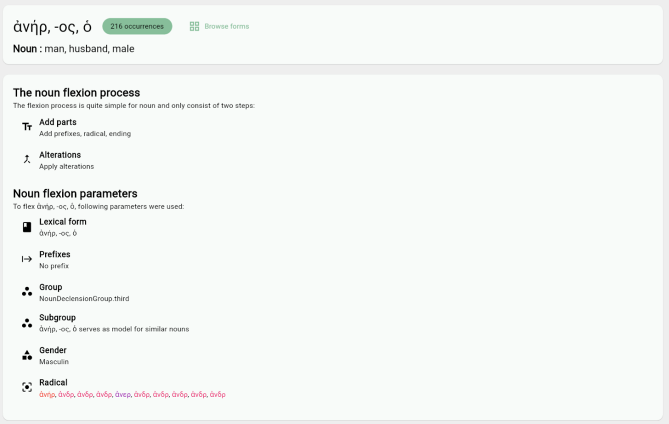
	Present			Aorist			Future			Perfect			Plus perfect			Imperfect		
	Active	Middle	Passive	Active	Middle	Passive	Active	Middle	Passive	Active	Middle	Passive	Active	Middle	Passive	Active	Middle	Passive
Indicative	λέγ...	λέγ...	λέγ...	⚠ εἶπ...	⚠ εἶπ...	λέγ...	λέγ...	λέγ...	λέγ...	⚠ εἰρηκ...	⚠ εἰρη...	⚠ εἰρη...	⚠ εἰρηκ...	⚠ εἰρη...	⚠ εἰρη...	λέγ...	λέγ...	λέγ...
Subjunctive	λέγ...	λέγ...	λέγ...	εἶπ...	εἶπ...	λέγ...	-	-	-	-	-	-	-	-	-	-	-	-
Imperative	λέγ...	λέγ...	λέγ...	εἶπ...	εἶπ...	λέγ...	-	-	-	-	-	-	-	-	-	-	-	-
Infinitive	λέγ...	λέγ...	λέγ...	⚠ εἶπ...	⚠ εἶπ...	λέγ...	-	-	-	⚠ εἰρηκ...	⚠ εἰρη...	⚠ εἰρη...	-	-	-	-	-	-
Participle	λέγ...	λέγ...	λέγ...	⚠ εἶπ...	⚠ εἶπ...	λέγ...	-	-	-	⚠ εἰρηκ...	⚠ εἰρη...	⚠ εἰρη...	-	-	-	-	-	-
Optative	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Radical:
- εἶπ, εἶπ, εἶπ, εἶπ, εἶπ, εἶπ

⚠ Instead of radical:
- λέγ, λέγ, λέγ, λέγ, λέγ, λέγ

Figure 12. Verb specification exploration. The radical is shown for each form.

Nouns and adjectives are quite similar, but the specification object is much simpler, so the tables are avoided. The figure 13 shows the specification of the noun ἀγάπη (love). Here key insights are the pattern made by the radical, highlighted in different colors, and the subgroup, that specify the model followed by the noun or the adjective.



ἀνήρ, -ος, ὁ

Noun : man, husband, male

The noun flexion process

The flexion process is quite simple for noun and only consist of two steps:

Add parts
Add prefixes, radical, ending

Alterations
Apply alterations

Noun flexion parameters

To flex ἀνήρ, -ος, ὁ, following parameters were used:

Lexical form
ἀνήρ, -ος, ὁ

Prefixes
No prefix

Group
NounDeclensionGroup.third

Subgroup
ἀνήρ, -ος, ὁ serves as model for similar nouns

Gender
Masculin

Radical
ἀνήρ, ἀνδρ, ἀνδρ, ἀνδρ, ἀνερ, ἀνδρ, ἀνδρ, ἀνδρ, ἀνδρ, ἀνδρ

Figure 13. Noun specification exploration. The radical is shown for each case.

V. Future works

A. Model improvements

While our grammar model demonstrates remarkable progress, certain limitations, already discussed in subsection I-E2, warrant future investigation and refinement. Our ongoing efforts will be directed towards addressing these limitations and expanding the model's capabilities.

A primary focus for enhancement involves the integration of contextual factors into our inflection prediction process. By accounting for the influence of neighboring words and the phonological shifts that arise within sentence structures, we aim to enhance the model's accuracy and applicability to real-world language usage.

The intricacies of accent placement also beckon for deeper exploration. While our model adeptly handles breathing marks, further work is required to refine its treatment of accentuation. Our intent is to delve into the intricate phonological rules governing accent placement, enabling the model to predict accentual variations more comprehensively and accurately.

Furthermore, we aspire to broaden the model's horizons beyond its current scope of the New Testament corpus. By considering a wider range of Koine Greek texts, we aim to enhance the model's generalization capabilities. This expansion would allow the model to tackle a more diverse array of linguistic constructs, rare words, and contextual variations encountered across different Koinè texts.

In the pursuit of these endeavors, we remain committed to refining and expanding our grammar model to better capture the complexities of Ancient Greek linguistics.

B. A learning mobile app

Looking ahead, we are excited to announce our plans for the development of a language learning application. This application will leverage the robust capabilities of our formal grammar model to offer a comprehensive and interactive learning experience. Our team aims to commence development by the end of 2023, with the intention of making the app available to universities by mid-2024. This initiative reflects our commitment to enhancing language education by incorporating advanced linguistic insights into a user-friendly and accessible platform. Through this endeavor, we aspire to provide students and educators with a valuable tool that facilitates the study and understanding of Ancient Greek, contributing to the broader landscape of language pedagogy.

VI. Conclusion

In this paper, we presented a formal grammar model designed to address the challenges of inflectional complexity in Ancient Greek. By focusing on the flexion of verbs, nouns, adjectives, and other word categories, our model demonstrates a significant improvement in accuracy through iterative manual curation and automatic optimization. Notably, our approach showcases remarkable performance enhancements for nouns and adjectives, while also highlighting the intricate nature of verb flexion and the importance of tailored exception handling.

While our model offers a promising foundation for understanding and generating accurate word forms, it is crucial to recognize its current limitations, including the exclusion of contextual factors and certain phonological intricacies. As we look to the future, our research opens avenues for refining and expanding our formal grammar model, with ongoing efforts to incorporate contextual influences, improve accent handling, and extend the model's applicability beyond the New Testament corpus to other Koine Greek texts.

Furthermore, we anticipate the development of a language learning application that capitalizes on the strengths of our formal grammar model. By providing students and educators with a powerful tool, we aim to enhance the study of Ancient Greek and contribute to the broader landscape of language education.

In summary, this work represents a significant step towards unlocking the complexities of Ancient Greek inflection and offers a platform for further research and practical applications. Our journey continues as we strive to refine and extend our model's capabilities, ultimately fostering a deeper understanding of this rich and historically significant language.

References

[1] Greek to english lexicon of biblical words. https://biblicaltext.com/dictionary.download/. We thank BiblicalText.com for providing free licensing of their resource.

[2] New testament greek text with parsing. https://bereanbible.com/bsb_tables.xlsx. We thank the Berean Bible Society for providing free licensing of their resource.

[3] A. Djaballah and M. Lafleur. Grammaire grecque du nouveau testament. Unpublished course material, Faculté de théologie évangélique, Université Acadia, Montréal.

[4] Scott Fleischman. Modeling the morphophonology of ancient greek. https://github.com/scott-fleischman/greek-grammar, 2015.