Natural Language Engineering (2024), **30**, pp. 1277–1300
doi:10.1017/S135132492300058X

ARTICLE

Automated annotation of parallel bible corpora with cross-lingual semantic concordance

Jens Dörpinghaus[1,2,3]

[1]Federal Institute for Vocational Education and Training (BIBB), Bonn, Germany, [2]University of Bonn, Bonn, Germany, and [3]University of Koblenz, Mainz, Germany
Email: jens.doerpinghaus@bibb.de

(Received 9 November 2022; revised 24 November 2023; accepted 20 December 2023;
first published online: 25 January 2024)

Abstract

Here we present an improved approach for automated annotation of New Testament corpora with cross-lingual semantic concordance based on Strong's numbers. Based on already annotated texts, they provide references to the original Greek words. Since scientific editions and translations of biblical texts are often not available for scientific purposes and are rarely freely available, there is a lack of up-to-date training data. In addition, since annotation, curation, and quality control of alignments between these texts are expensive, there is a lack of available biblical resources for scholars. We present two improved approaches to the problem, based on dictionaries and already annotated biblical texts. We provide a detailed evaluation of annotated and unannotated translations. We also discuss a proof of concept based on English and German New Testament translations. The results presented in this paper are novel and, to our knowledge, unique. They show promising performance, although further research is needed.

Keywords: Corpus annotation; cross-lingual; parallel corpora; biblical texts; Strong's numbers

1. Introduction

Building a concordance of texts, automated text alignment, and automated text translation are well-studied research topics. A *semantic concordance* is a widely used approach to link text corpora with data and values in lexicons (see Landes, Leacock, and Tengi 1998). Even in the humanities, much research has been done in this broad field of text mining and automated text processing. Coming to the field sometimes called *Digital Theology* as a subfield of *Digital Humanities* and its intersection with ancient languages, we still see several challenges, although the problems themselves may seem simple and a standard task.

Here we want to address the challenge of automatically annotating words within New Testament texts in order to create parallel Bible corpora in different languages. Our goal is to create cross-lingual concordances for New Testament texts and translations. These are widely used for research and teaching, see Fig. 1 for a typical use case.

Thus, the research problem can be stated as follows: Given a Bible text in English or German language, how can we annotate the corresponding Greek or Hebrew word in the original text? Usually this annotation is done using Strong's numbers which we will introduce in the next Section. For example, in John 4:4 ("And he had to pass through Samaria") the word "and" should be annotated by the Strong's number G1161 referring to Greek $\delta\varepsilon$ and "Samaria" to G4540. Thus, the task is to assign of Strong's numbers to bible versions which currently do not have these numbers.

© The Author(s), 2024. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives licence (http://creativecommons.org/licenses/by-nc-nd/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided that no alterations are made and the original article is properly cited. The written permission of Cambridge University Press must be obtained prior to any commercial use and/or adaptation of the article.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Figure 1. Illustration of a parallel bible view provided by `https://www.stepbible.org`. It shows two English translations (ESV and KJV) and a Greek text (SBLG).

Our approach is limited to the mapping between translated words, given both the translation with or without further information, and the Greek source with morphological information. This paper is a revised, improved, and extended version of Dörpinghaus and Düing (2021). Here, it was shown that AI approaches based on CRFs do not perform as well as rule-based algorithmic approaches, for example, with an F_1-score of 0.13 compared to 0.84 for Luther 1912. However, this work had several challenges, such as a lack of detailed analysis and limited evaluation on a few already annotated texts. We will present improvements that lead to a well-functioning environment for some use cases: First, we generalized the approach to work with all available POS categories. Second, we have improved the algorithm with a generic function to enhance different values (e.g., by the position of words in a sentence) and a generic threshold that helps to analyze different scenarios. Third, we evaluate different strategies to limit the number of words that can be assigned to categories. Finally, we present an evaluation of several texts.

Research on biblical texts and translations has a long tradition, and translations have been widely used. In the nineteenth century, there was a great increase in the number of different Bible translations, and thus research in this field also increased (see Metzger 2001).

New approaches from computer science have also been used to evaluate translations and texts, but have only really taken off in the last 30 years as they have become more accessible to scholars with different backgrounds. It is possible to use these methods to understand the manual curation and understanding of texts. For example, automated identification of actors and locations can help to understand (social) networks in literary texts (see Dörpinghaus 2021). In addition, these methods may also be useful for other early Christian texts and may help to improve the automated processing of, for example, ancient church orders or biblical apocrypha. In addition, these methods can help improve the technological solutions for automated approaches. First, it is possible to use these methods to understand the manual curation and understanding of the text. Second, it would be possible to improve the technological solutions for automated approaches. Here, Clivaz (2017) states that very little research has been done in this area. Anderson (2018) underlines the lack of interest of theologians in digital and modern text mining methods a year later. The field of digital theology is emerging, but shows more interest in current trends in digitization (see Sutinen and Cooper 2021). For a detailed discussion of this topic (see also Dörpinghaus 2022). Only the areas of digital manuscripts, digital academic research, and publishing show some progress (see Clivaz, Gregory, and Hamidović 2013). This work tries to be a step toward closing this gap.

Because scholarly editions and translations of biblical texts are often not freely available, and because annotation, curation, and quality control of alignments between these texts are expensive, there is a lack of available biblical resources for scholars. The goal of this work is to develop and evaluate novel approaches for automatically generating alignments for parallel Bibles, leading to cross-lingual semantic concordance. We have based our work on the Sword Project,[a] which

[a]See https://crosswire.org/sword/.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

provides a full API available under the GNU license, and has been able to work with Greek texts, English and German translations available.

In this work, we present an improved approach for automated annotation of New Testament corpora with cross-lingual semantic concordance based on Strong's numbers. We introduce two improved approaches to the problem, based on dictionaries and already annotated biblical texts. We provide a detailed evaluation of annotated and unannotated translations. We also discuss a proof of concept based on English and German New Testament translations. The results presented in this paper are novel and, to our knowledge, unique. They show promising performance, although further research is needed.

The remainder of the paper is organised as follows: The second section gives a brief overview over the state of the art and related work. The third section is dedicated to the data foundation. We will also discuss the annotation style and the selection of training and test data. In the fourth section, we present two approaches to tackle the problem. The fifth section is dedicated to experimental results on annotated and non-annotated translations. Our conclusions are drawn in the last section. The results presented in this paper are novel and, to the best of our knowledge, unique. They show promising performance, although further research is needed.

2. Related work

Since little research has been done in this area, we list all available materials, even if their tasks are only tangentially related. In biblical research, *The Exhaustive Concordance of the Bible* from 1890 is widely used to link words from biblical texts to dictionary entries. These so-called Strong's numbers can be used to create automatically aligned parallel texts, see Cysouw, Biemann, and Ongyerth (2007) or Wälchli (2010), who created semantic maps from parallel text data. Here, texts in several languages are presented together (see Simard 2020). It is important to note the discrepancy between other fields of research and the study of biblical texts: Although several approaches in biblical research are based on machine translation, these texts are still mainly hand-crafted, see for example the Greek-Hebrew-Finnish corpus of Yli-Jyrä *et al.* (2020) or the approach described by Rees and Riding (2009) and Riding and Steenbergen (2011). Even though the Bible is often used as a training or reference model for unsupervised learning models for translation, see for example Diab and Finch (2000), Resnik, Olsen, and Diab (1999), Christodouloupoulos and Steedman (2015), only few approaches have been made to analyze religious or theological texts with methods from AI and text mining.

For example, McDonald (2014) applied statistical methods to religious texts to evaluate their similarity based on word vectors. Another simple analysis was carried out by Verma (2017), and research on the reuse of historical texts was done by Büchler *et al.* (2010) using text mining technologies. Usually word frequencies are used to discuss the common authorship of biblical books (see e.g., Erwin and Oakes 2012). These so-called stylometric studies are not without critique (see Eder 2013).

To cover the linguistic question, other scholars have examined the impact of computer technology on Bible translation and discussed its limitations (see Riding 2008). Since Bible translations are not usually the subject of linguistic research, but are interesting for the history of languages, there is a wide range of publications and analyses of recent translations, see for example Renkema and van Wijk (2002) and De Vries (2000). There is also a considerable amount of literature on Bible translation (see Scorgie *et al.* 2009). It is important to note that Bible translation is not just a matter of choosing between translation strategies such as formal or dynamic equivalence.[b]

Encoding linguistic information in multilingual documents produces *Interlinear Glossed Text* (IGT). Biblical texts are usually well studied, so both Strong's numbers and morphological information are available for Hebrew and Greek texts. Automated glossing is also a widely studied

[b]For further details, we refer to Kerr (2011) or Metzger (2001).

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

area for other texts and languages (see Rapp, Sharoff, and Zweigenbaum 2016; McMillan-Major 2020; Zhao *et al.* 2020). Much work in this area has been devoted to methods based on neural networks and word embeddings. Sabet *et al.* (2020) applied static and contextualized embeddings and showed that an approach without parallel data or dictionaries could produce multilingual embeddings. However, their approach did not generalize to all languages in their test environment. Another approach was presented by Dou and Neubig (2021): They used fine-grained embeddings and parallel corpora. Their work is limited to several modern languages and generally shows comparable results to other models. Notably, the performance differs from one language to the other, showing that there is a lot of detailed work to be done depending on the particular language. Current AI approaches have the disadvantage that it is usually difficult to understand and adapt details in models. Therefore, we will pay special attention to translation approaches and their properties. Recently, Yousef *et al.* (2022a) not only introduced a gold standard for ancient Greek texts (to English and Portuguese) but also worked on tuning translation alignments by combining unsupervised training on mono- and bilingual texts with supervised training on manually aligned sentences. This clearly shows that large corpora of training data are very important (see also Palladino, Shamsian, and Yousef 2022). For biblical texts, many parallel texts are available. However, Koine Greek is different from other variants of Ancient Greek. Their work is accompanied by a tool for manual creation of alignment corpora, see Yousef *et al.* (2022b), and visual evaluation of models (see Yousef, Heyer, and Jänicke 2023). See also the survey by Sommerschield *et al.* (2023). AI-based approaches have also been used by Tyndale House in Cambridge to create parallel Bible corpora (see Instone-Brewer 2023). However, their work based on the Berkeley Word Aligner required 70 volunteers to complete the work. Problems with AI methods on biblical texts have also been identified by Dörpinghaus and Düing (2021). Only a little research has been done on the Qur'an (see Muhammad 2012). Some research has been done on word-for-word translation, especially for word-for-word translation without parallel data (see Conneau *et al.* 2017; Li *et al.* 2021). Other work has been done on misalignment (see Tsvetkov and Wintner 2012). For automated translation, there are no resources for ancient Greek (see Biagetti, Zanchi, and Short 2021). Other approaches, such as GASC (see Perrone *et al.* 2019), build a Bayesian model to describe the evolution of words and meanings in ancient texts. They note "a lack of previous works that focussed on ancient languages". Thus, not only are the target texts a new field, but we have very little work to build on in the field of automated translation.

In summary, the combination of different methods is the key to obtaining high-quality alignments, see for example Fei, Zhang, and Ji (2020), Steingrimsson, Loftsson, and Way (2021), Vu *et al.* (2021). It is a crucial point for the creation of interlinear glossed biblical texts to really understand the detailed concepts of the languages and either use large training corpora and supervise the results of these methods, for example, by manually curating the texts. Because of these various complexities, we decided to apply and improve classical algorithmic approaches to identify the underlying challenges.

3. Data

3.1. Overview

Here we will focus on the original Greek text and its representation in the English and German translations of the Bible, although this approach can be applied to any other language. There are several software packages available for accessing biblical texts. Some commercial software, such as Logos, provide no or very limited access to their API.[c] So we have based our work on the SWORD project, which provides a full API available under the GNU license.[d] We selected biblical texts

[c]See for example https://wiki.logos.com/Logos_4_COM_API.
[d]See http://crosswire.org/sword/index.jsp.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Table 1. Overview of training and test data. Here *tft* refers to thought-for-thought, *pa* to paraphrase approach, and *wfw* to word-for-word (formal equivalence). Texts with Strong's numbers are used for training and testing, and texts without Strong's numbers only for testing. The Remarks column indicates special cases: For the Leonberger Bible, translations based on two different Greek texts are available, and the VOLX-Bible provides a text based on the German colloquial youth language

Name	Language	Strong	Year	Approach	Remarks
Luther 1912	German	✓	1912	wfw and tft	
Luther 2017	German	–	2017	tft	
Leonberger Bible (GerLeoNA28/RP18)	German	✓	2017	wfw	NA28 and RP18
Schlachter (SLT)	German	–	2000	wfw	
Hoffnung für alle (HFA)	German	–	2002	pa	
VOLXBIBEL (VOLX)	German	–	2021	pa	Youth-language
King James Version (KJV)	English	✓	1769	wfw	
English Standard Version (ESV)	English	✓	2001	wfw	
American Standard Version (ASV)	English	✓	1901	wfw	
New Revised Standard Version (NRSV)	English	–	1989	pa	
World English Bible (WEB)	English	–	2015	pa	

```
<w lemma="strong:G2532" savlm="strong:G2532">And</w> he
<w lemma="strong:G3004" savlm="strong:G3004">said</w> to
<w lemma="strong:G0846" savlm="strong:G0846">him</w>
```

Figure 2. A snippet of the XML output for Acts 1:1 from diatheke.

based on their availability under an open license that ensures reproducibility and diverse translation approaches. For the Greek text, we used the SBLGNT 2.0 from Tyndale House, based on SBLGNT v.1.3 from Crosswire. This text is comparable to the Nestle-Aland/United Bible Societies text with some minor changes. The English texts are based on the KJV (King James Version, 1769), ASV (American Standard Version, 1901), and ESV (English Standard Version, 2011). The German texts are based on Luther (1912), Leonberger Bible (2017, based on Nestle-Aland 28 or Robinson-Pierpont 18). All data are available under a free license.[e]

There are several approaches to translating biblical texts. The KJV, ESV, and ASV follow a traditional word-for-word approach, also known as formal equivalence. The Leonberger Bible follows the same approach, while the Luther 1912 also includes elements of the thought-for-thought approach known as dynamic equivalence. For testing purposes, we will also consider translations that use a paraphrase approach. We will use the New Revised Standard Version (NRSV), the World English Bible (WEB), Luther 2017, Hoffnung for alle (HFA), and later the very free text of the German VOLXBIBEL. See Table 1 for an overview. For a detailed overview of Bible translations (see Metzger 2001).

There are several annotations that can be displayed in different ways. Here we rely on XML output.[f] Both lemmatical and morphological information are contained in w-tags. For an example on Acts 1:1, see Fig. 2.

[e]See http://www.crosswire.org/sword/modules/ for details on these packages.
[f]This option is called HTML output format by the used software diatheke, see Section 4.1.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

However, additional morphological information may be available: <w lemma="strong:G3588" morph="robinson:T-ASM" savlm="strong:G3588" src="21">τον</w>. They are usually stored according to RMAC (Robinson's Morphological Analysis Codes see Robinson 1973). Thus, we will use the existing morphological information, if available. If not, we will proceed as described in the next section. In summary, we will use this XML-based annotation style for extracting information as well as for storing and comparing data.

3.2. Training and test data

To collect the training data, we can use the complete New Testament texts mentioned above. This results in 7957 verses in each version. There are 5624 entries in Strong's dictionary. We tested our models on a complete corpus, or a random subset of both the same and different translations (see Dörpinghaus 2023) for data. Comparing fully annotated texts is easy because we can use the whole corpus. That is, we computed precision, recall, and F_1-score for annotating Strong's numbers on the entire New Testament corpus. Here, the corresponding gold standards are available as Sword modules.[g]

To evaluate non-annotated texts, we created gold standards for several verses and translations. Some of these will be discussed in more detail because they show certain limitations of our approach. However, to evaluate the quality measures, we selected 20 verses from different books, both from the Gospels and the Acts of the Apostles, and from different epistles. It was important to select a variety of verses, both from narrative texts (Mk 10:3; Lk 1:9; Jn 12:2, 21:1; Acts 8:14) and from Gospel-specific verses (Mark 1:1; John 19:35), enumerations (Acts 27:5), apocalyptic texts (Rev 1:19; 14:5), and letters (Rom 1:1; 12:4; Eph 1:8; 1 Peter 1:10; 1 John 1:5; Jude 1:8). To make our approach comparable to other methods, we have published this gold standard (see Dörpinghaus 2023).

In addition, we will test our model on some verses from newer versions, such as the new German VOLX Bible. Here the verses are evaluated manually. For a detailed overview, see Table 1.

4. Methodology

4.1. Workflow

All steps have been implemented using `SWORD` 1.9.0.3874,[h] `diatheke` 4.8[i] as CLI front end, and `Python` 3.8. We used the following libraries: `BeautifulSoup`[j] for XML parsing, `spaCy`[k] for POS tagging and `jellyfish`[l] for measuring the difference between two strings, for example, by Levenshtein distance. Using different texts from `SWORD` and different language models in `spaCy` shows that we can easily switch the language-specific components. Thus, at least for similar input and output languages, the proposed workflow could in principle be language independent. However, our examples are based on English and German texts, which only shows that this assumption holds for Germanic languages. The extent to which this holds for other languages needs to be investigated.

[g]See https://crosswire.org/sword/modules/.
[h]See https://crosswire.org/sword/.
[i]See https://wiki.crosswire.org/Frontends:Diatheke.
[j]See https://www.crummy.com/software/BeautifulSoup/bs4/.
[k]See https://spacy.io/.
[l]See https://jamesturk.github.io/jellyfish/.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

4.2. Modeling

We have biblical texts that contain verses. Each verse X contains a sequence of words, so

$$X^L = x_1^L, \ldots, x_N^L \tag{1}$$

$$X^{L'} = x_1^{L'}, \ldots, x_M^{L'}. \tag{2}$$

However, without loss of generality, let us assume that L contains annotations to Strong's numbers. Then we want to model the target glossing $f : X^L \to X^{L'}$, which contains mappings from a word origin $x_i^L \in X^L$ to another word $x_j^{L'} \in X^{L'}$ with the same Strong's number. Let Y be a sequence of all mappings, then we have to compute $P(Y|X^L)$.

We need to add a short note about verses in biblical texts: Verse content and numbering can differ slightly, for example, Catholic Bible texts use a different numbering especially for Old Testament texts (see Mayer and Cysouw 2014). However, all considered texts use the same numbering scheme, and in any case, the Sword Library can handle these different schemes.

Due to the amount of annotated data for biblical texts, we have several options:

- If L and L' are different languages, we must find the appropriate syntactic equivalent in L'. This can be complicated, especially for certain grammatical constructions that have a different form in L'. For example, if L is ancient Greek, it is unclear whether this approach will work for languages such as English or German. Also, there are no language models for Ancient Greek or Hebrew, see Dörpinghaus and Düing (2021) and the survey provided by Sommerschield *et al.* (2023).
- If L and L' are the same language and annotated texts exist, the task is reduced to finding the match for a given part of speech. However, this is only true for syntactically close translations and may have several other restrictions, for example, for varieties of languages.

Here we propose a two-step method. As input, we use the target text (a translated text) verse by verse, if necessary, the original Greek text with Strong's annotations and some additional information from dictionaries and biblical translations. Then we annotate the target glossing. See Fig. 3 for an illustration.

4.3. Preprocessing and matching

After detecting parts of speech in the target text, we can sort words from the original annotated source text and the target text based on parts-of-speech. This helps to reduce the target word set. Since we know the Greek Strong's numbers, we can use lemmatization to compare words and assign the best match. A first algorithmic approach was presented in Dörpinghaus and Düing (2021), see Algorithms 1. Here, the authors preprocessed with a parts-of-speech tagger limited to a list of five categories: nouns, verbs, conjunctions, prepositions, and pronouns. We will call this approach POS_0. However, we did not change the libraries used for parts-of-speech recognition. This is due to the fact that spaCy shows one of the best results on current German texts, and there is very little difference in performance between spaCy and other libraries like NLTK or StanfordNLP (see Ortmann, Roussel, and Dipper 2019). In addition, we wanted to compare the former and the new method. However, the underlying algorithm does not depend on the library used for parts-of-speech detection.

In this paper, we will also apply an approach that covers all parts-of-speech detected by `spaCy`,[m] which we will denote by POS_1. We get a set

$$\mathfrak{P} = \{POS_0, POS_1\}.$$

[m] See the official `SpaCy` documentation and source code available at https://github.com/explosion/spaCy/blob/master/spacy/glossary.py for a detailed overview.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Algorithm 1 Dictionary-based-matches I

Require: Sequences of words $X^L = x_1^L, \ldots, x_N^L$ with dictionary mapping to $dict(x_i^L)$ to dictionary Δ and in target language $X^{L'} = x_1^{L'}, \ldots, x_M^{L'}$.

Ensure: Mapping $f : X^L \rightarrow X^{L'}$.

```
for c in POS do
    for x_i^L in c do
        v = []
        for x_j^L' in c do
            v ← δ(lem(dict(x_i^L)), lem(x_j^L'))
        end for
        find x_j^L' by min v
        assign f(x_i^L) = x_j^L'
    end for
end for
return f
```

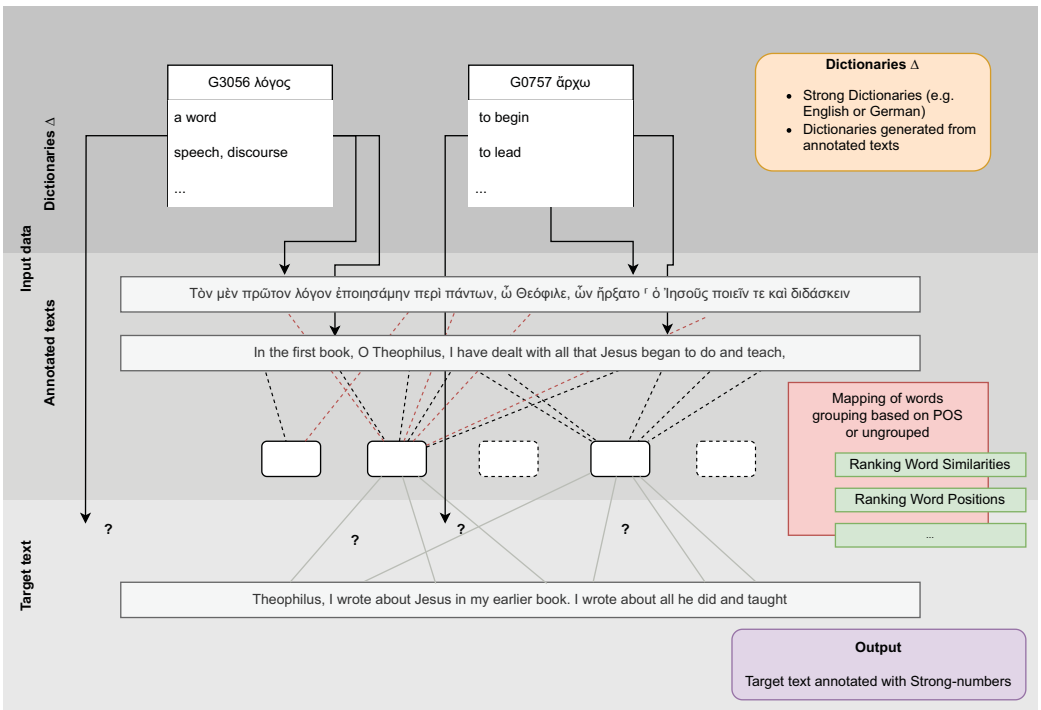

Figure 3. The proposed method with example data (Acts 1:1). In general, we use as input the target text (a translated text) verse by verse and an existing annotated text. The original Greek text with annotations could be used when adding a translation or dictionary. Existing dictionaries can be used, or dictionaries can be created from the use of Strong's numbers in a given translation. First, we use POS tagging and lemmatization to extract the matching words. Then we annotate the target gloss by finding the best matches, either by grouping words at POS or by considering all available terms.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Algorithm 2 Dictionary-based-matches II

Require: Sequences of words $X^L = x_1^L, \ldots, x_N^L$ with dictionary mapping to $d(x_i^L)$ to dictionary D and in target language $X^{L'} = x_1^{L'}, \ldots, x_M^{L'}$.

Require: Threshold ε

Require: A set of functions f

Ensure: Mapping $f : X^L \to X^{L'}$.

```
    for c in POS do
2:    for x_i^L in c do
        v = []
4:      for x_j^L' in c do
          R = f(lem(d(x_i^L)), lem(x_j^L'))
6:        v ← δ(lem(d(x_i^L)), lem(x_j^L')) + R
        end for
8:      find x_j^L' by x = min v
        if x < ε then
10:         assign f(x_i^L) = x_j^L'
        end if
12:   end for
    end for
14: return f
```

Thus, POS_1 and POS_0 differ in the number of parts of speech considered for matching. However, in these approaches, matches are found only within one category, for example, only verbs are matches, see for example line 1, in Algorithm 2. However, we may modify this approach, since the usage of conjunctions, prepositions, and pronouns is not consistent across languages, and may even vary within a single language. We denote the mixing of all parts-of-speech by *all* and the mixing of only conjunctions, prepositions and pronouns by *cpp*. We get a set

$$\mathfrak{C} = \{all, none, cpp\}.$$

Thus, each approach is a triple set defining input, parts-of-speech approach, and matching approach: $(input, \mathfrak{p} \in \mathfrak{P}, \mathfrak{c} \in \mathfrak{C})$. We will use $*$ to indicate that we will consider multiple approaches for this element, for example $(bible, POS_1, *)$ includes all approaches in $\mathfrak{C}$. See Table 2 for an overview.

In the Algorithms 1 and 2, the function δ refers to a distance function (such as Levenshtein distance or cosine similarity). The function *dict* returns dictionary entries for a given word, which can be used for mapping between different languages. However, in the case of equal languages, we define $dict(w) = w$. To distinguish between dictionary-based and a dictionary extracted directly from the original text, we use Δ as input when relying on one or more dictionaries.

In our case, we extracted all the translations used by a particular Bible for a given Strong's number, since Dörpinghaus and Düing (2021) showed several difficulties when working with existing dictionaries. Although further research could be done using resources such as the lexical-semantic network for German "GermaNet" (see Kunze and Wagner 2001), or the lexical database "WordNet" (see Miller 1995). Thus, the main difference between a Bible and Δ as input is that the former matches only lemmata from a particular source, while Δ matches all usages of a particular Strong's number within the source.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Table 2. Overview of the approaches evaluated in this paper. *POS* and POS_0 differ in the number of parts of speech considered for matching. The column "categories" describe whether only elements within a category are matched (all), whether all elements are mixed (none), or whether only conjunctions, prepositions, and pronouns are mixed (cpp)

(*input*, ꝑ, ꞓ)	POS	Categories	Input
(*bible*, POS_0, *all*)	POS_0	all	Input Bible
(*bible*, POS_1, *all*)	POS_1	all	Input Bible
(*bible*, POS_1, *none*)	POS_1	none	Input Bible
(*bible*, POS_1, *cpp*)	POS_1	mixing conj, prep, pron	Input Bible
(Δ, POS_1, *all*)	POS_1	all	Input Dictionaries Δ

Algorithm 3 Extract Dictionary

Require: Bible Text b with Strong's-Numbers
Ensure: Dictionary

```
    d = []
2:  for s in (1, 5625) do
        ev = find("G" + s)
4:      for v in ev do
            w = getWords(v, "G" + s)
6:          d["G" + s] ← lemma(w)
        end for
8:  end for
    return d
```

To make this data available, we wrote an importer that creates a list of words in the target language that are associated with a Strong's number. This dictionary-based approach is a lazy learner approach, since we learn the dictionaries first, but the comparison and mapping are done in a separate step.

However, it is clear that this approach can be optimized. Dörpinghaus and Düing (2021) showed several weaknesses of this naive approach. For example, even for the same source and target corpus, the results were not perfect. In line 8 of the Algorithm 1, we can define a threshold ε and adjust line 6 to reinforce special cases, for example, the order of words.

Thus, in Algorithm 2, we introduce a set of functions f to reinforce different values. For example, we can replace $f = pos_z(x, y)$ with

$$pos_z(x, y) = \begin{cases} z & pos(x) = pos(y) \\ 0 & else \end{cases}$$

to reinforce labels that are at a similar position in the source and target text.

4.4. Extracting dictionaries

In Algorithm 3, we present an approach to extract dictionaries from annotated translations. Given a Bible text b with Strong's numbers, we iterate over all 5625 Greek terms in line 2. The function

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

find in line 5 returns all verses containing a given Strong's number. In line 3, we use the function *getWords* which returns the lemmatized usage of a Strong's number within a given verse.

The result highlights the specification of certain translations. For example, G0033 ("age", used only in James 4:13; 5:1) is not annotated by the Leonberger Bible, while Luther 1912 annotates the term "Wohlan". Other translations are different, for example, Luther 1912 translates G0001 as "A" and Leonberger Bible as "Alpha". ASV and ESV translate G0032 similarly with "angel" and "messenger". However, the ASV does not annotate G0029, while the ESV uses "force" and "compel".

This already foreshadows some problems we will discuss in the next section.

4.5. Evaluation

The performance of the approaches is evaluated by comparing each annotation in its final output with the test dataset of annotated biblical texts. Thus, we need to cross-evaluate different input scenarios against different and similar output scenarios.

Since our approach produces Strong's numbers annotations for words in the translated text, the first question is whether this leads to correct assignments on the *same* text. We will also evaluate whether combining different models leads to better solutions. Since these approaches may predict Strong's numbers that have more or fewer occurrences in the text, we add both precision and recall to our evaluation, defined as follows:

$$Recall = \frac{TP}{FN + TP} \tag{3}$$

$$Precision = \frac{TP}{FP + TP} \tag{4}$$

Here TP means true positives (a correct assignment), FN false negatives (assigning no or the wrong Strong's number to a word which originally has one), and FP false positives (assigning a Strong's number to a word which does not have one). Thus, $FP + TP$ returns all positive results and $FP + TP$ are all samples that should have been identified as positive. The F_1 score is the harmonic mean of *Precision* and *Recall*. The best value is 1 and the worst value is 0. The formulas used are

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}, \tag{5}$$

These metrics are presented as a micro-average across all verses. Furthermore, we will analyze how these systems work on unannotated translations. For this purpose, a few verses have been selected to evaluate the output.

5. Results

5.1. Results on POS_0

We evaluated and compared the output of Algorithm 2 with $\varepsilon = 2$ and $f = pos_z(x, y)$ for Luther 1912 and GerLeoNA28 as target texts and Luther 1912, GerLeoRP18 and GerLeoNA28 as sources. See Table 3 for a detailed overview. Here, the same target and source texts lead to a significantly better result than the results presented in Dörpinghaus and Düing (2021). Most interestingly, the F_1 score is high not only for the same input and output but also for the other texts. Thus, the Leonberger Bible text seems to be very close to Luther, both syntactically and in word choice. However, we could also reproduce that the Leonberger Bible as target text behaves differently: Here, the combined approach improves the results, while this is not the case for Luther 1912.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Table 3. Results of algorithm $(bible, POS_0, all)$ with $\varepsilon = 2$ and $f = pos_z(x, y)$ for Luther 1912 and GerLeoNA28 as target texts. The column "F_1 (D)" shows the results by Dörpinghaus and Düing (2021)

Luther 1912					GerLeoNA28				
Base	Prec.	Recall	F_1	F_1 (D)	Base	Prec.	Recall	F_1	F_1 (D)
Luther 1912	**0.917**	**0.846**	**0.88**	0.84	Luther 1912	0.75	0.549	0.634	0.55
GerLeoNA28	0.854	0.768	0.809	0.78	GerLeoNA28	0.864	0.76	0.809	**0.67**
GerLeoRP18	0.831	0.74	0.783	–	GerLeoRP18	**0.896**	0.822	0.857	–
Combined	0.844	0.77	0.805	**0.86**	Combined	0.88	**0.854**	**0.867**	**0.67**

Table 4. Results of algorithm $(bible, POS_0, all)$ with $f = pos_z(x, y)$ for KJV ($\varepsilon = 2$) and ESV ($\varepsilon = 16$) as target texts

KJV					ESV				
Base	Prec.	Recall	F_1	F_1 (D)	Base	Prec.	Recall	F_1	F_1 (D)
KJV	**0.727**	**0.75**	**0.738**	**0.58**	KJV	0.422	0.905	0.576	0.74
ESV	0.554	0.679	0.61	0.46	ESV	**0.554**	**0.962**	**0.703**	**0.78**
ASV	0.535	0.688	0.602	0.53	ASV	0.496	0.944	0.651	0.75
Combined	0.5	0.691	0.58	0.49	Combined	0.538	0.961	0.69	**0.78**

Table 5. Results of algorithm $(bible, POS_0, all)$ with $\varepsilon = 2$ and $f = pos_z(x, y)$ for Luther 2017 and HFA as target texts

Luther 2017				HFA			
Base	Precision	Recall	F_1 score	Base	Precision	Recall	F_1 s core
Luther 1912	0.984	0.729	**0.838**	Luther 1912	**1.0**	0.753	**0.859**
GerLeoNA28	**0.898**	0.717	0.797	GerLeoNA28	0.904	0.763	0.827
GerLeoRP18	0.869	0.712	0.783	GerLeoRP18	0.878	0.763	0.816
Combined	0.894	**0.747**	0.814	Combined	0.884	**0.777**	0.827

Comparing these results with the English translations, KJV and ESV, in Table 4, reveals some interesting observations. While our approach significantly improves for the KJV compared to Dörpinghaus and Düing (2021), it performs worse for the ESV, which also follows a word-for-word approach. However, while the advantage of this evaluation is the existence of a fully annotated text, it provides a very specific environment.

In order to analyze the results of previously unannotated texts, we created a gold standard for several verses from the Gospels, Acts, and Epistles for several translations. A detailed evaluation with precision, recall, and F_1-score can be found in Table 5, for the German translations HFA, a more free translation, and Luther 2017, which is close to Luther 1912. Again, the results are much better than Dörpinghaus and Düing (2021).

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Table 6. Results of algorithm (*bible*, POS_0, *all*) with $\varepsilon = 2$ and $f = pos_z(x, y)$ for NRSV and WEB as target texts

NRSV				WEB			
Base	Prec.	Recall	F_1 score	Base	Prec.	Recall	F_1 score
KJV	0.853	0.509	0.637	KJV	0.84	0.404	0.545
ESV	**0.915**	0.641	0.754	ESV	**0.891**	0.462	0.609
ASV	0.843	0.645	0.73	ASV	0.784	0.463	0.582
Combined	0.857	**0.703**	**0.772**	Combined	0.796	**0.556**	**0.655**

The dictionary-based approaches on German translations (Table 5) show very promising results. The precision value is high, although the recall value increases for HFA and Leonberger Bible. We see a different behavior for Luther 1912 and GerLeoNA28. For the latter, the combination of both dictionaries increases the recall, but also decreases the precision. This means that a smaller proportion of data is correctly annotated, but the relative proportion of correctly annotated data increases. This implies that the amount of annotated parts strongly depends on the data used – it is not as simple as "more is better", but it is crucial to note that a combination of dictionaries needs to be carefully investigated.

One of the reasons may be that although both translations were done with the same approach, there are more than a hundred years between them considering Luther 1912. Thus, the words and their meanings may have changed. In the next section, we will make some preliminary observations about more recent translations.

This is even more significant for the evaluation of the English translations in Table 6. ESV and ASV are both based on the KJV, and again there are more than a hundred years between them (1769, 1901, 2011). The two most recent translations show a good result, the recall value is high, and the precision value increases with the matching dictionary. The most remarkable result can be found when using KJV for a combination of dictionaries, it even decreases the values. This result has further strengthened our confidence that it is crucial to evaluate the dictionary base for this approach.

However, as we have already mentioned, the precision value is also misleading, since we only evaluate the words recognized by the POS-tagger approach. So we can see two extreme situations: First, the translation has added several phrases and words. So we have *more* words to tag than words in the original Greek text. Second, the translation uses different paraphrases and constructs to express a longer Greek text, resulting in *less* words to tag than words in the original Greek text. In Table 7, we have evaluated all six texts we used for our tests. In most cases, there are more words in the original text than our POS tagging engine could detect. For the German texts, the average is well below one, but the extreme values have a higher amount. However, the values are exactly the same for each language. So we can summarize that all available texts had a similar approach.

As we can see, we have selected different values for the bound ε. In Figs. 4 and 5, we provide a detailed analysis of the F_1 scores for changing ε. As we can see, there is an optimal value, but it has to be found by experiment. Usually, the F_1 score does not improve significantly when $\varepsilon > 3$. However, we must emphasize the importance of preprocessing the texts, even for annotated texts. The KJV annotates texts differently, for example, in Acts 1:3, we find the following annotations of multi-word statements and phrases, while other translations like ESV annotate single words, see Fig. 6.

This leads to problems when using KJV as a basis for further annotations. Thus, a further improvement might consider more extensive preprocessing of previously annotated texts.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Table 7. This table shows the minimum, average, and maximum difference between the total number of references to Greek words (Strong's numbers) and the detected number of words. Interestingly, these numbers are the same for all texts in one particular language

Bible text	min	avg	max
KJV	−5	5.833333	25
ESV	−5	5.833333	25
ASV	−5	5.833333	25
Luther 1912	−11	0.895833	27
GerLeoNA28	−11	0.895833	27
GerLeoRP18	−11	0.895833	27

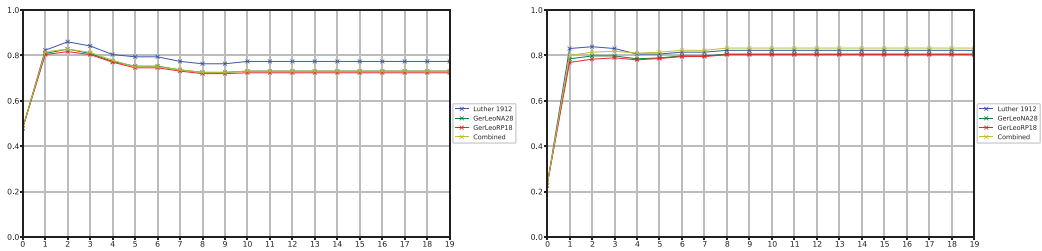

Figure 4. F_1 Score for different values of ε (x-axis) for HFA (left) and Luther 2017 (right) as target text.

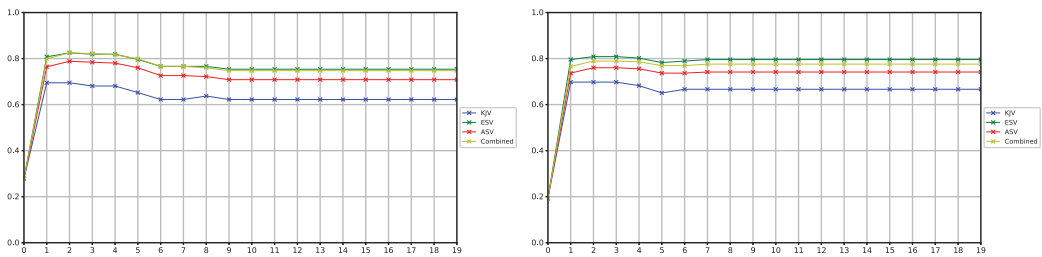

Figure 5. f_1 Score for different values of ε (x-axis) for NRSV(left) and WEB (right) as target text.

```
['G5039', 'infallible proofs'],
['G3700', 'being seen'],
['G4012', 'of the things pertaining to']
```

```
['G5039', 'proofs'],
['G3700', 'appearing'],
['G4012', 'about']
```

Figure 6. Example of KJV (top) and ESV (bottom) annotations for Acts 1:3.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

G11	ϟ	G2464	G2464	ϟ	G2384	G2384	ϟ	G2455
Abraham	zeugte	Isaak.	Isaak	zeugte	Jakob.	Jakob	zeugte	Juda
Abraham	*begat*	*Isaac.*	*Isaac*	*begat*	*Jacob.*	*Jacob*	*begat*	*Judah*
G2532		G80						
und	seine	Brüder.						
and	*his en*	*brethren.*						

Figure 7. Application to Luther 2017 (Matthew 1:2). The corresponding English text according to the ASV is: "Abraham begat Isaac, and Isaac begat Jacob, and Jacob begat Judah and his brethren."

G11			ϟ		G2464		G2464		ϟ	
Abraham	war	der	Vater	von	Isaak.	Auf	Isaak		folgten	
Abraham	*was*	*the*	*father*	*of*	*Isaac.*		*Isaac*		*was followed by*	
				ϟ		ϟ		G2455	G2532	
in direkter		Linie		Jakob –	der	Vater	von	Juda	und	seinen
in direct		*line of ancestors*		*Jacob,*	*the*	*father*	*of*	*Judah*	*and*	*his*
G80	G2455									
Brüdern –,	Juda									
brothers,	*Judah.*									

Figure 8. Application to HFA (Matthew 1:2). The corresponding English text according to the ASV is: "Abraham begat Isaac, and Isaac begat Jacob, and Jacob begat Judah and his brethren."

5.2. Testing on non-annotated translations

To test our approach on a recent translation, we will use different verses with different linguistic challenges. First, we will use both Luther 1912 and GerLeoNA28 as a basis for mapping. As a first example, we will consider Matt 1:2, which is a noun-centered sentence. For Luther 2017, we get the assignment shown in Fig. 7.

While previous approaches assign G1161 ($\delta\varepsilon$) instead of G2532 ($\kappa\alpha\iota$), indicating the challenge of assigning the correct particles, the proposed approach works correctly. This assignment contains 2 missing assignments of Strong's numbers. This text is identical to the 2006 Elberfelder translation.

We will show the performance on two more German translations, following a thought-for-thought approach known as dynamic equivalence. Hoffnung für alle (HFA, 2015) is less rigorous than the VOLXBIBEL (2014), which follows a youth communication paradigm. The results in Fig. 8 were run with $\varepsilon = 4$. It contains 3 wrong or missing assignments. All verbs are missing. Again, particles are a challenge, the $\delta\varepsilon$ of the original Greek sentence is missing; but this is also due to the fact that it was omitted in the translation. The results done with $\varepsilon = 6$ only show additional misclassified attributes, "Vater" (father) was wrongly assigned to G2384.

Here the description is changed. Instead of describing the begetter, a passive construction ". . .is father of. . ." was chosen. The word father was assigned, but could not be found in the Greek text. These errors increase when this method is applied to the VOLXBIBEL, see Fig. 9.

There are many paraphrases (e.g. mixing son and father, both wrongly assigned), additional terms (the promise of land), and additional words ("Leute", "Land", etc.). Thus, some assignments are neither truly correct nor incorrect. For example, the translation uses "und" (and) for both $\kappa\alpha\iota$ and $\delta\varepsilon$, while the algorithm assigns only G2532. Other particles are mostly missing. However, most names are assigned correctly. In previous work, not a single word is correctly assigned (see Dörpinghaus and Düing 2021). In summary, our approach works best for formal equivalence or dynamic equivalence translations. While it will not work for paraphrase approaches. Here

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Table 8. Results of algorithm (*bible*, POS_1, *all*) with $\varepsilon = 6$ and $f = pos_z(x, y)$ for Luther 1912 and GerLeoNA28 as target texts

Luther 1912				GerLeoNA28			
Base	Prec.	Recall	F_1 score	Base	Prec.	Recall	F_1 score
Luther 1912	**0.917**	**0.971**	**0.943**	Luther 1912	0.589	0.914	0.716
GerLeoNA28	0.8	0.893	0.844	GerLeoNA28	0.732	0.95	0.827
GerLeoRP18	0.758	0.862	0.806	GerLeoRP18	**0.789**	0.956	**0.865**
Combined	0.756	0.87	0.809	Combined	0.766	**0.959**	0.852

Table 9. Results of algorithm (*bible*, POS_1, *all*) with $\varepsilon = 6$ and $f = pos_z(x, y)$ for Luther 2017 and SLT as target texts

Luther 2017				SLT			
Base	Precision	Recall	F_1 score	Base	Precision	Recall	F_1 score
Luther 1912	**0.57**	**0.92**	**0.704**	Luther 1912	**0.602**	**0.903**	**0.722**
GerLeoNA28	0.487	0.865	0.623	GerLeoNA28	0.507	0.849	0.635
GerLeoRP18	0.459	0.843	0.594	GerLeoRP18	0.476	0.823	0.603
Combined	0.487	0.858	0.621	Combined	0.492	0.834	0.619

G11 G2455↯ G2464↯ ↯ ↯ G2532 ↯
Matt. 1:2 (VOLX) *Zuerst war da Abraham, dem Gott ein großes Land für sich und seine*
G2455↯ ↯ G11 G2464↯ G2464 G2455↯ G2384 G2384
Leute versprochen hatte. Abrahams Sohn war Isaak, Isaaks Sohn war Jakob, Jakob war der
G2384↯ ↯ G2455 G2532 G80
Vater von Juda und dessen Brüdern.

Figure 9. Assignment on VOLX in Matt. 1:2 (*Abraham begat Isaac; and Isaac begat Jacob; and Jacob begat Judah and his brethren*).

only some parts can be annotated, for example, nouns (locations, names, etc.) or certain verbs. Although their defining characteristic is that they do not match the original language word for word, this annotation is still useful for linking to encyclopedias or other cross-references.

5.3. Results on POS_1

As we discussed above, the translation uses different paraphrases and constructs to express a longer Greek text, which results in fewer words to be tagged than in the original Greek text. For German translations, this value is between −19 and 16 (average −4.791) and for English translations between −13 and 0 (average −5.375). So the situation is different from the results of POS_0 – overall we have fewer words in the original text than our POS-tagger detected.

Tables 8 and 9 show the results for (*bible*, POS_1, *all*) for German translations. For Luther 1912, the results are comparable, although the recall values are higher. The same is the case for Luther 2017 and SLT. Here the F_1 score is lower than with the (*bible*, POS_0, *all*) approach. In Fig. 10, we

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

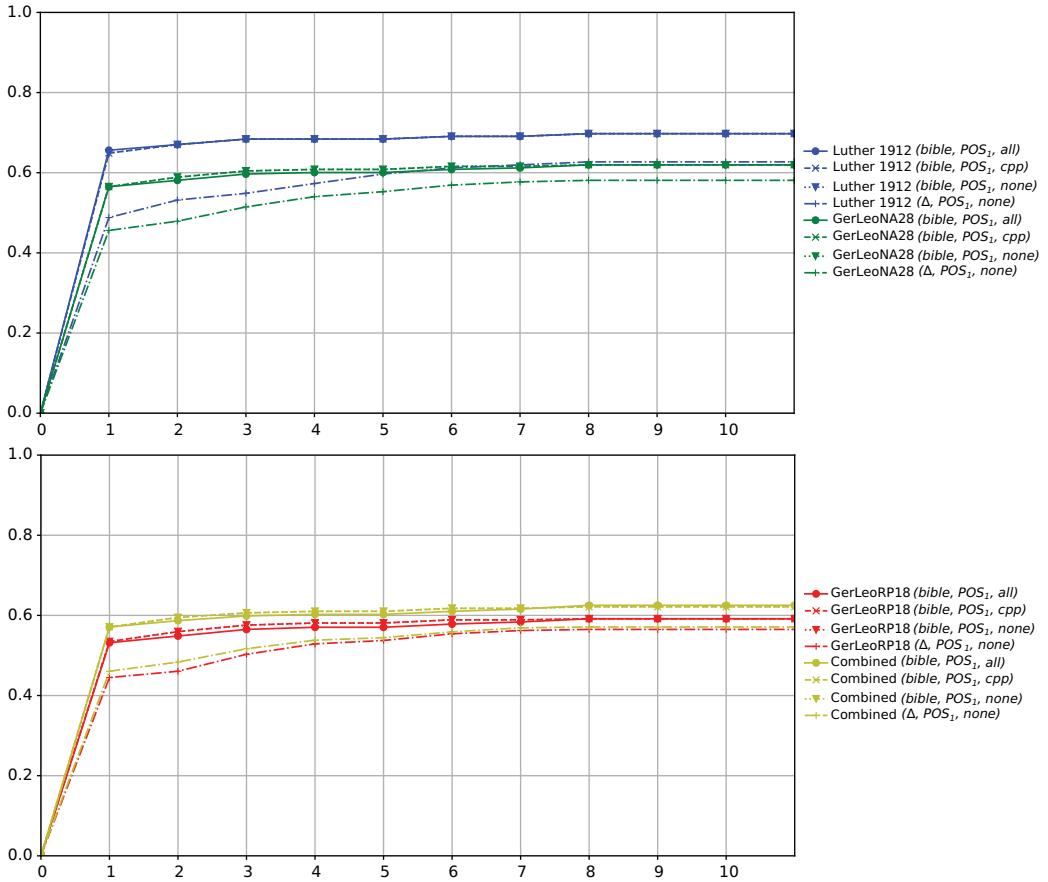

Figure 10. f_1 Score for different values of ε (x-axis) for Luther 2017 as target text.

show the output of (*bible*, POS_1, *all*), (*bible*, POS_1, *cpp*), (*bible*, POS_1, *none*), and (Δ, POS_1, *all*) for different translations as input. We see that (*bible*, POS_1, *all*) is the best approach overall, and the choice of dictionaries is key. Combining different dictionaries does not improve the output in general, but might be a good choice if the best input is unknown.

Tables 10 and 11 show the results for (*bible*, POS_1, *all*) on English translations. Here, (*bible*, POS_1, *all*) provides better results than (*bible*, POS_0, *all*) for KJV and ESV and comparable results for NRSV and WEB. Again, recall is generally higher. Table 11 also shows some surprises. First, larger values for ε worsen the F_1 score for NRSV. Second, for WEB, it neither improves nor decreases the F_1 score, but while precision increases, recall decreases. In Figs. 11 and 12, we show the output of (*bible*, POS_1, *all*), (*bible*, POS_1, *cpp*), (*bible*, POS_1, *none*), and (Δ, POS_1, *all*) for different translations as input.

However, Table 10 shows a significant performance improvement for $\varepsilon = 13$. While all other approaches do not change at about $\varepsilon > 9$, the KJV is special, as we discussed earlier. For example, in Romans 20:5, POS_1 finds 39 parts of speech, while only 14 Strong's numbers are assigned, see Fig. 13.

This explains why a higher value of ε still increases the F_1 score. However, it also underlines the need for a detailed understanding of the texts and the annotation of Strong's numbers.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Table 10. Results of algorithm (*bible*, POS_1, *all*) with $f = pos_z(x, y)$ for KJV ($\varepsilon = 13$) and ESV ($\varepsilon = 8$) as target texts

KJV				ESV			
Base	Precision	Recall	F_1 score	Base	Precision	Recall	F_1 score
KJV	**0.938**	**1.0**	**0.968**	KJV	0.472	0.85	0.607
ESV	0.708	**1.0**	0.829	ESV	**0.676**	**0.926**	**0.781**
ASV	0.596	0.983	0.742	ASV	0.611	0.912	0.732
Combined	0.554	0.986	0.71	Combined	0.596	0.922	0.724

Table 11. Results of algorithm (*bible*, POS_1, *all*) and $f = pos_z(x, y)$ for NRSV ($\varepsilon = 2$) and WEB as target texts

NRSV				WEB			
Base	Precision	Recall	F_1 score	Base	Precision	Recall	F_1 score
KJV	0.714	0.6	0.652	KJV	**0.718**	0.622	0.667
ESV	**0.781**	0.728	0.754	ESV	0.764	0.747	0.756
ASV	0.714	0.729	0.722	ASV	0.669	0.74	0.703
Combined	0.756	**0.764**	**0.76**	Combined	0.699	**0.757**	**0.727**

5.4. Testing on non-annotated translations

Again, to test our approach on a recent translation, we will use different verses with different linguistic challenges. First, we will use both Luther 1912 and GerLeoNA28 as a basis for assignment. As a first example, we will consider Matt 1:2, which is a noun-centered sentence. For Luther 2017, we get the assignment in Fig. 14.

POS_1 recognizes the preposition "sein" (*autos*), but does not assign G0846 to it. In general, G1161 ($\delta\varepsilon$) is missing, but is also omitted in the translation. Overall, the results are better than POS_0. We will show the performance of two more German translations, following a thought-for-thought approach known as dynamic equivalence. Hoffnung für alle (HFA, 2015) is less rigorous than the VOLXBIBEL (2014), which follows a youth communication paradigm. The results in Fig. 15 were run with $\varepsilon = 4$.

The assignment of G1080 for "folgen" (to follow) is a paraphrase of "beget". This result shows more parts of speech than POS_0, overall the quality is comparable. We can see that thought-for-thought approaches are challenging. These errors increase when this method is applied to the VOLXBIBEL, see Fig. 16.

Again, there are a lot of paraphrases (e.g. mixing son and father, both wrongly assigned), additional terms (the promise of land), and additional words ("Leute", "Land", etc.). This means that some assignments are neither correct nor incorrect, similar to POS_0. Again, the translation uses "und" (and) for both $\kappa\alpha\iota$ and $\delta\varepsilon$, while only G2532 is assigned by the algorithm. In addition, POS_1 finds more parts of speech, but does not assign Strong's numbers to all of them. While most of the additional verbs are wrong, some assignments are missing ("der", "ein", "für", etc.).

6. Discussions and conclusions

6.1. Summary

This paper describes two improved approaches for automatically annotating words within New Testament texts to create parallel Bible corpora in different languages based on (*bible*, POS_0, *all*)

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

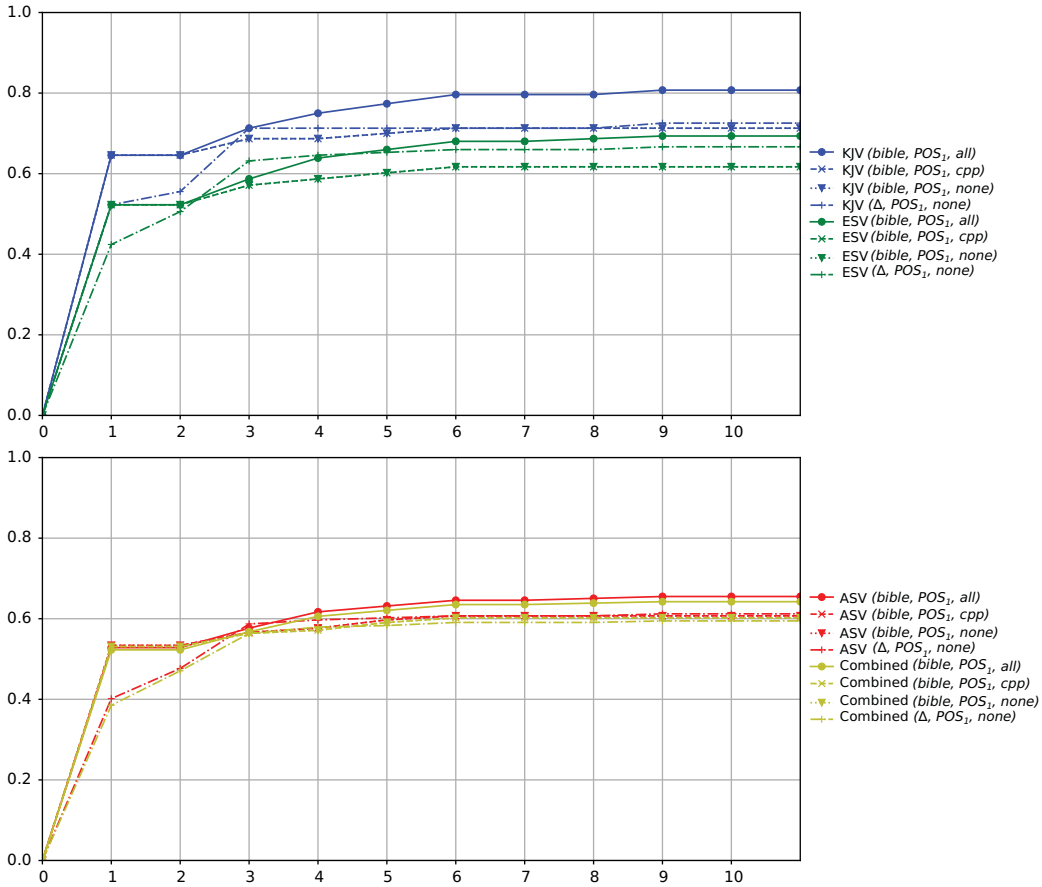

Figure 11. F_1 Score for different values of ε (x-axis) for KJV as target text.

and (*bible*, POS_1, $*$). Automated annotation of words within biblical texts to create parallel biblical corpora in different languages for cross-lingual concordance alignment of New Testament texts and translations is still an important research topic. On the one hand, it is a limited problem with a fixed set of texts, and on the other hand, it is challenging because it relies on Ancient Greek and Hebrew. We proposed a lazy learner approach using dictionaries of existing annotations and dictionaries extracted from annotated texts. However, our approach emphasizes the importance of proper preprocessing of the data, handling morphology and phrases.

Another contribution of this work is the publicly available evaluation dataset, which can be used to make further work in this area comparable.

Although the amount of training data was generally limited due to strict licensing policies in the field of theology, we were able to obtain promising results for some translations. Applying this approach to thought translations does not seem reasonable, but limiting this approach to nouns can provide links to encyclopedias with good quality.

6.2. Limitations

This paper covers many overlapping fields, such as linguistics, Bible translation, ancient languages, theology, NLP, and ML. Thus, we find several limitations in the output with respect to domain. Some of them are discussed here.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

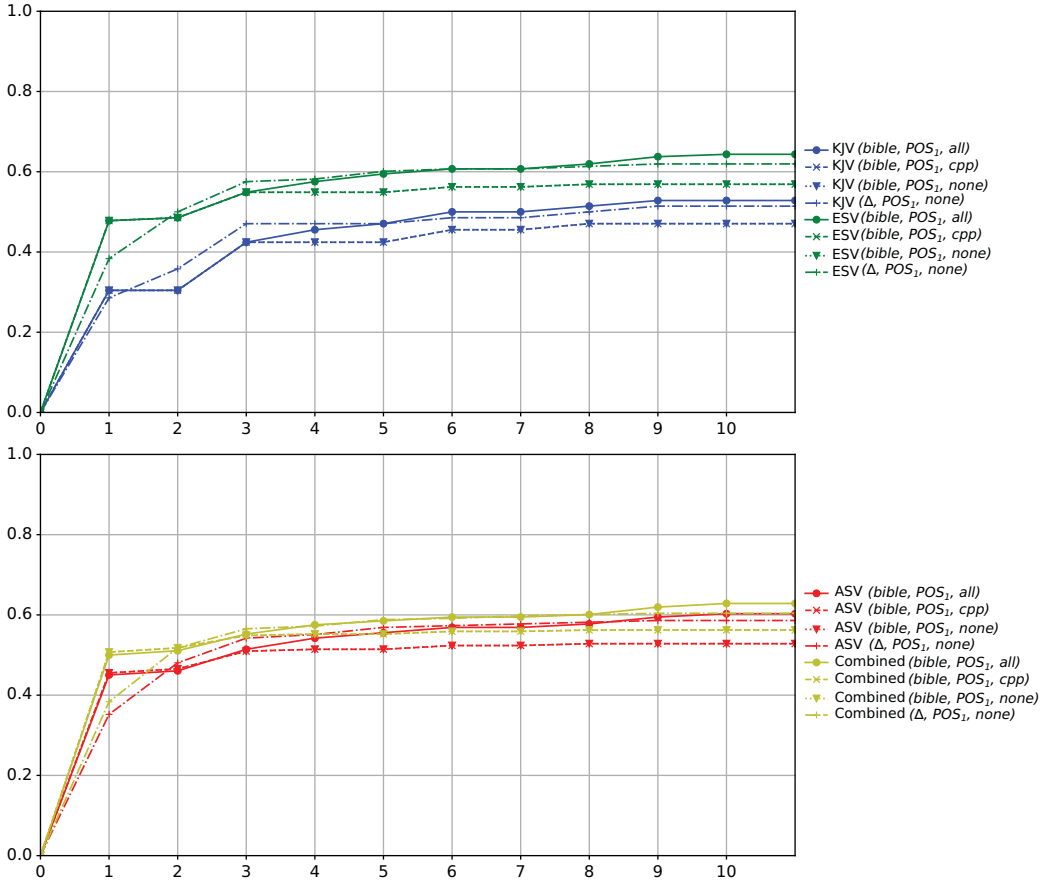

Figure 12. F_1 Score for different values of ε (x-axis) for ESV as target text.

```
['G5461', 'will bring to light'],
['G5319', 'will make manifest']
```

```
['', 'will', 14,71,4],
['', 'bring', 15,76,5],
['', 'to', 16,82,2],
['', 'light', 17,85,5]
```

Figure 13. Example of existing (top) and POS_1 (bottom) annotations for Romans 20:5.

G11	G1080	G2464	G2464	G1080	G2384	G2384	G1080	G2455
Abraham	zeugte	Isaak.	Isaak	zeugte	Jakob.	Jakob	zeugte	Juda
Abraham	*begat*	*Isaac.*	*Isaac*	*begat*	*Jacob.*	*Jacob*	*begat*	*Judah*
G2532		G80						
und	seine	Brüder.						
and	*his en*	*brethren.*						

Figure 14. Application to Luther 2017 (Matthew 1:2). The corresponding English text according to the ASV is: “Abraham begat Isaac, and Isaac begat Jacob, and Jacob begat Judah and his brethren.”

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

G11	G846ϟ		G2464ϟ		G2464		G2464		G1080	
Abraham	war	der	Vater	von	Isaak.	Auf	Isaak		folgten	
Abraham	*was*	*the*	*father*	*of*	*Isaac.*		*Isaac*		*was followed by*	
	G2384ϟ		G2384		G80ϟ		G2455	G2532		
in direkter	Linie		Jakob –	der	Vater	von	Juda	und	seinen	
in direct	*line of ancestors*		*Jacob,*	*the*	*father*	*of*	*Judah*	*and*	*his*	
G80	G2455									
Brüdern –,	Juda									
brothers,	*Judah.*									

Figure 15. Application to HFA (Matthew 1:2). The corresponding English text according to the ASV is: "Abraham begat Isaac, and Isaac begat Jacob, and Jacob begat Judah and his brethren."

							G846ϟ		*G11*		*G2464ϟ*			*G2464ϟ*	*ϟ*	*ϟ*	*G2532*	*ϟ*
Matt. 1:2 (VOLX)	*Zuerst*	*war*	*da*	*Abraham,*	*dem*	*Gott*	*ein*	*großes*	*Land*	*für*	*sich*	*und*	*seine*					
G2388ϟ	*ϟ*	*G1080ϟ*	*G11*	*G2464ϟ*	*ϟ*	*G2464*		*G2455ϟ*	*G846ϟ*	*G2384*	*G2384*	*G486ϟ*					
Leute	*versprochen*	*hatte*	*.*	*Abrahams*	*Sohn*	*war*	*Isaak,*	*Isaaks*	*Sohn*	*war*	*Jakob,*	*Jakob*	*war*				
G2384ϟ	*ϟ*	*G2455*	*G2532*		*G80*												
der	*Vater*	*von*	*Juda*	*und*	*dessen*	*Brüdern.*											

Figure 16. Assignment on VOLX in Matt. 1:2 (*Abraham begat Isaac; and Isaac begat Jacob; and Jacob begat Judah and his brethren*).

- Different approaches to Bible translation present different challenges. While our approach works better for formal equivalence, it is limited for paraphrase approaches. In general, since other AI approaches have been shown not to perform as well as rule-based algorithmic approaches, it remains unclear how well approaches for automated annotation of parallel Bible corpora will perform.
- Since there are no language models for ancient Greek or Hebrew (see Dörpinghaus and Düing 2021), we are limited in the use of AI methods. On the other hand, our results could be more useful if they could contribute to existing models. Other languages require further discussion.
- While the approaches do not assign Strong's numbers that are not used within a verse, they do not necessarily assign correct numbers to a part of speech. We provide extensive experimental results on mixing different POS, but limiting to one category seems most reasonable. For other languages, especially non-Germanic languages, this may not be the case. In particular, we could not provide an in-depths analysis of errors for POS-tagging. While our results can be used by laypeople, or provide an initial foundation for later expert curation, they cannot be used in the field of theology without further restrictions.
- In other words: Our evaluation was done on English and German translations. This is a serious limitation, as both are Germanic languages. How does the approach work in other languages? Further experiments could help to discuss the usefulness of this approach for other languages. In addition, it would be valuable to test this approach on languages with fewer resources.

6.3. Future work

Here, we presented an improved method for the automated annotation of parallel Bible corpora work with Strong's numbers, providing a cross-lingual semantic concordance. We introduced a

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

pipeline that uses the SWORD API. Our approach yields results that depend on the input data and the translation approach of the target Bible. For word-for-word translations, it provides a highly accurate baseline that could be used for further expert curation. However, this method cannot be applied to translations that follow a paraphrase approach, such as the German VOLXBIBEL, and it shows lower performance for non-word-by-word approaches. As noted above, it is also questionable whether it is useful to apply this approach to paraphrased texts beyond linking to encyclopedia entries for nouns. However, this work will hopefully lead to further research and a better understanding of the special requirements in the field of theology, especially in ancient languages.

Our analysis of the limitations reveals a number of questions and possible further improvements:

- First, we need to consider whether more translations, dictionaries, synonyms, and biblical texts can be used as training data. Although recall may not always improve when more dictionaries are used, a better data basis combined with improvements in modeling and algorithms will improve the results.
- Second, we need to investigate our approach to parts of speech, because we found that the number of POS and Strong's numbers in a verse varies. Thus, the gap between the Strong's annotations in the original texts and the POS tagged words needs to be closed.
- Third, the proposed approach does not depend on the library used for POS detection. Since we were able to identify some errors using POS detection, we suggest further research on the performance of other libraries such as StanfordNLP or NLTK.
- Finally, an in-depth error analysis should be done for other AI approaches such as the CRF models presented in other papers. Here, it should be analyzed whether a better feature selection (e.g. POS tagging or dependency labels) is the key.

While our proof of concept is both working and generic, it is still early work on a problem that needs more attention. It already provides useful output for several use cases. In other cases, it could help to automatically build a foundation for detailed manual annotation of texts. However, we hope that it will also highlight the importance of more interdisciplinary research in this field.

References

Anderson C. (2018). Digital humanities and the future of theology. *Cursur_: Zeitschrift für Explorative Theologie*.

Biagetti E., Zanchi C. and Short W.M. (2021). *Toward the creation of wordnets for ancient indo-european languages*. In *Proceedings of the 11th Global Wordnet Conference*, pp. 258–266.

Büchler M., Geßner A., Eckart T. and Heyer G. (2010). Unsupervised detection and visualisation of textual reuse on ancient greek texts.

Christodouloupoulos C. and Steedman M. (2015). A massively parallel corpus: the bible in 100 languages. *Language Resources and Evaluation* **49**(2), 375–395.

Clivaz C. (2017). Die Bibel im digitalen Zeitalter: Multimodale Schriften in Gemeinschaften. *Zeitschrift für Neues Testament* **39**(40), 35–57.

Clivaz C., Gregory A. and Hamidović D. (2013). *Digital Humanities in Biblical, Early Jewish and Early Christian Studies*. Leiden and Boston: Brill.

Conneau A., Lample G., Ranzato M.A., Denoyer L. and Jégou H. (2017). Word translation without parallel data. arXiv preprint arXiv:1710.04087.

Cysouw M., Biemann C. and Ongyerth M. (2007). Using strong's numbers in the bible to test an automatic alignment of parallel texts. *STUF-Language Typology and Universals* **60**(2), 158–171.

De Vries L. (2000). Bible translation and primary orality. *The Bible Translator* **51**(1), 101–114.

Diab M. and Finch S. (2000). A statistical word-level translation model for comparable corpora. Technical report, University of Maryland Institute for Advanced Computer Studies.

Dou Z.-Y. and Neubig G. (2021). Word alignment by fine-tuning embeddings on parallel corpora. arXiv preprint arXiv:2101.08231.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Dörpinghaus J. (2021). Die soziale netzwerkanalyse: neue perspektiven für die auslegung biblischer texte? *Biblisch Erneuerte Theologie* **5**, 75–96.

Dörpinghaus J. (2022). Digital theology: new perspectives on interdisciplinary research between the humanities and theology. *Interdisciplinary Journal of Research on Religion* **18**, 1–17.

Dörpinghaus J. (2023). Evaluation Data for the Annotation of German and English New Testament Texts with Strong's Numbers. URL https://doi.org/10.5281/zenodo.8024803.

Dörpinghaus J. and Düing C. (2021). Automated creation of parallel bible corpora with cross-lingual semantic concordance. In *2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS)*. IEEE, pp. 111–114.

Eder M. (2013). Computational stylistics and biblical translation: how reliable can a dendrogram be. *The Translator and the Computer*, 155–170.

Erwin H. and Oakes M. (2012). Correspondence analysis of the new testament. In *Workshop Organizers*, p. 30.

Fei H., Zhang M. and Ji D. (2020). Cross-lingual semantic role labeling with high-quality translated training corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7014–7026.

Instone-Brewer D. (2023). Computational alignment of Greek and Hebrew with Bible translations, using Swahili as a proof of concept. Available at https://docs.google.com/presentation/d/1XgTRMsvQ-55W2nUmZ1aQQ4a2aYR56O_17ZUi0ZiLaNM/edit#slideid.g2878d53056_2_75 (accessed: 25 May 2023).

Kerr G.J. (2011). Dynamic equivalence and its daughters: placing bible translation theories in their historical context. *Journal of Translation* **7**(1), 13–20.

Kunze C. and Wagner A. (2001). Anwendungsperspektiven des GermaNet, eines lexikalischsemantischen Netzes für das Deutsche. *Chancen und Perspektiven Computergestützter Lexikographie* **107**, 229–246.

Landes S., Leacock C. and Tengi R.I. (1998). Building semantic concordances. *WordNet: An Electronic Lexical Database* **199**(216), 199–216.

Li Y., Zhang Y., Yu K. and Hu X. (2021). Adversarial training with wasserstein distance for learning cross-lingual word embeddings. *Applied Intelligence* **51**(11), 1–13.

Mayer T. and Cysouw M. (2014). Creating a massively parallel Bible corpus. *Oceania* **135**(273), 40.

McDonald D. (2014). A text mining analysis of religious texts. *The Journal of Business Inquiry* **13**(1), 27–47.

McMillan-Major A. (2020). Automating gloss generation in interlinear glossed text. *Proceedings of the Society for Computation in Linguistics* **3**(1), 338–349.

Metzger B.M. (2001). *The Bible in Translation: Ancient and English Versions*. Biblical Studies. Baker Publishing Group.

Miller G.A. (1995). WordNet: a lexical database for English. *Communications of the ACM* **38**(11), 39–41.

Muhammad A.B. (2012). *Annotation of Conceptual Co-Reference and Text Mining the Qur'an*. University of Leeds.

Ortmann K., Roussel A. and Dipper S. (2019). Evaluating off-the-shelf NLP tools for German. In *KONVENS*

Palladino C., Shamsian F. and Yousef T. (2022). Using parallel corpora to evaluate translations of ancient greek literary texts. an application of text alignment for digital philology research. *Journal of Computational Literary Studies* **1**(1), 703–747.

Perrone V., Palma M., Hengchen S., Vatri A., Smith J.Q. and McGillivray B. (2019). GASC: genre-aware semantic change for Ancient Greek, Association for Computational Linguistics. In *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change, Florence, Italy*, pp. 56–66. https://www.aclweb.org/anthology/W19-4707.

Rapp R., Sharoff S. and Zweigenbaum P. (2016). Recent advances in machine translation using comparable corpora. *Natural Language Engineering* **22**(4), 501–516. https://doi.org/10.1017/S1351324916000115.

Rees N. and Riding J. (2009). Automatic concordance creation for texts in any language. *Proceedings of Translation and the Computer* **31**, 1–11.

Renkema J. and van Wijk C. (2002). Converting the words of God: an experimental evaluation of stylistic choices in the new Dutch bible translation. *Linguistica Antverpiensia, New Series–Themes in Translation Studies* **1**, 169–190.

Resnik P., Olsen M.B. and Diab M. (1999). The bible as a parallel corpus: annotating the book of 2000 tongues. *Computers and the Humanities* **33**(1), 129–153.

Riding J.D. (2008). Statistical glossing, language independent analysis in bible translation. *Translating and the Computer* **30**, 703–747.

Riding J. and Steenbergen G. (2011). Glossing technology in paratext 7. *The Bible Translator* **62**(2), 04–102. https://doi.org/10.1177/026009351106200206.

Robinson H. (1973). *Morphology and Landscape*. University Tutorial Press.

Sabet M.J., Dufter P., Yvon F. and Schütze H. (2020). Simalign: high quality word alignments without parallel training data using static and contextualized embeddings. arXiv preprint arXiv:2004.08728.

Scorgie G.G., Strauss M.L., Voth S.M., et al. (2009). *The Challenge of Bible Translation: Communicating God's Word to the World*. Zondervan Academic.

Simard M. (2020). Building and using parallel text for translation. In *The Routledge Handbook of Translation and Technology*, pp. 78–90.

Sommerschield T., Assael Y., Pavlopoulos J., Stefanak V., Senior A., Dyer C., Bodel J., Prag J., Androutsopoulos I. and de Freitas N. (2023). Machine learning for ancient languages: a survey. *Computational Linguistics*, 1–44.

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press

Steingrimsson S., **Loftsson H. and Way A.** (2021). CombAlign: a tool for obtaining high-quality word alignments. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pp. 64–73.

Sutinen E. and Cooper A.-P. (2021). *Digital Theology: A Computer Science Perspective*. Emerald Group Publishing.

Tsvetkov Y. and Wintner S. (2012). Extraction of multi-word expressions from small parallel corpora. *Natural Language Engineering* **18**(4), 549–573. https://doi.org/10.1017/S1351324912000101.

Verma M. (2017). Lexical analysis of religious texts using text mining and machine learning tools. *International Journal of Computer Applications* **168**(8), 39–45.

Vu T., **He X.**, **Phung D. and Haffari G.** (2021). Generalised unsupervised domain adaptation of neural machine translation with cross-lingual data selection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3335–3346.

Wälchli B. (2010). Similarity semantics and building probabilistic semantic maps from parallel texts. *Linguistic Discovery* **8**(1), 331–371.

Yli-Jyrä A., **Purhonen J.**, **Liljeqvist M.**, **Antturi A.**, **Nieminen P.**, **Räntilä K.M. and Luoto V.** (2020). HELFI: a Hebrew-Greek-Finnish Parallel Bible Corpus with Cross-Lingual Morpheme Alignment. arXiv preprint arXiv:2003.07456.

Yousef T., **Heyer G. and Jänicke S.** (2023). Evalign: visual evaluation of translation alignment models. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pp. 277–297.

Yousef T., **Palladino C.**, **Shamsian F.**, **d'Orange Ferreira A. and dos Reis M.F.** (2022a). An automatic model and gold standard for translation alignment of ancient greek. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 5894–5905.

Yousef T., **Palladino C.**, **Shamsian F. and Foradi M.** (2022b). Translation alignment with ugarit. *Information-an International Interdisciplinary Journal* **13**(2), 65.

Zhao X., **Ozaki S.**, **Anastasopoulos A.**, **Neubig G. and Levin L.** (2020). Automatic interlinear glossing for under-resourced languages leveraging translations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 5397–5408.

Cite this article: Dörpinghaus J (2024). Automated annotation of parallel bible corpora with cross-lingual semantic concordance. *Natural Language Engineering* **30**, 1277–1300. https://doi.org/10.1017/S135132492300058X

https://doi.org/10.1017/S135132492300058X Published online by Cambridge University Press