Deriving Consensus for Multi-Parallel Corpora: an English Bible Study

Patrick Xia and **David Yarowsky**
Center for Language and Speech Processing, Johns Hopkins University
{paxia, yarowsky}@jhu.edu

Abstract

What can you do with multiple noisy versions of the same text? We present a method which generates a single consensus between multi-parallel corpora. By maximizing a function of linguistic features between word pairs, we jointly learn a single corpus-wide multiway alignment: a consensus between 27 versions of the English Bible. We additionally produce English paraphrases, word-level distributions of tags, and consensus dependency parses. Our method is language independent and applicable to any multi-parallel corpora. Given the Bible's unique role as alignable bitext for over 800 of the world's languages, this consensus alignment and resulting resources offer value for multilingual annotation projection, and also shed potential insights into the Bible itself.

1 Introduction

Noisy or heterogeneous copies of the same text are prevalent in religious and literary texts (Resnik et al., 1999; Koppel et al., 2016), machine translation n-best lists (Kumar and Byrne, 2004; Papineni et al., 2002), comparable corpora (Barzilay and Lee, 2003), and social media (Xu et al., 2015). While copies can be analyzed independently or together in a pairwise manner, information can be lost by not using them all jointly.

We view these copies of text as **multi-parallel corpora**, which consist of multiple sets of comparable or partially aligned documents. This contrasts with parallel corpora, which are usually between only two. The goal of this work is to produce word alignments for multi-parallel corpora (Fig. 1).

We approach this problem by tying the multi-parallel corpora together using features such as pairwise word alignments, dependency parses, and POS tags. Our method jointly learns word alignments and annotations for these features in the English Biblical multi-parallel corpora. We produce multiway word alignments, complete dependency parses, and POS tag annotations for the English Bible. While our resources and choice of features are catered for our specific domain, the method can be applied more broadly for aligning and establishing consensus in any domain.

Consensus	newsimplified	montgomery	lexham
Then	(Then, 0)	(Thereupon, 0)	(Then, 0)
Herod	(Herod, 1)	(Herod, 1)	(Herod, 1)
secretly	(secretly, 2)	(secretly, 3)	(secretly, 2)
called	(called, 3)	(sent, 2)	(summoned, 3)
for		(for, 4)	
the	(the, 4)	(the, 5)	(the, 4)
wise		(wise, 5)	
men	(astrologers, 5)	(Magi, 6)	(men, 6)

Figure 1: A sample of Fig. 3, in which different words with a similar meaning are aligned. Each entry contains the word and its index in the original sentence.

The English Bible is a literary religious text with multiple authors, disputed authorship structure, and multiple revisions for language modernization.[1] While there is existing computational work in Biblical analysis (Lee, 2007), our contribution of automatically generated consensus annotations for all verses allows future research to efficiently investigate across all English Bibles. As the Bible is available in electronic form in over 800 of the world's languages (Mayer and Cysouw, 2014), the Bible may be the only parallel corpus for low-resource languages, and our in-domain resources can be a valuable reference.[2]

[1] The unresolved Synoptic Problem questions the order and dependencies of the the Synoptic Gospels.

[2] Available at github.com/pitrack/monolign.

Proceedings of the The 8th International Joint Conference on Natural Language Processing, pages 448–453,
Taipei, Taiwan, November 27 – December 1, 2017 ©2017 AFNLP

2 Method

The consensus consists of aligned tokens between the corpora. Assuming each corpus consists of partially aligned documents (e.g. verses of the Bible), we first target the word alignments at the document level. We then use a bootstrapping approach to produce the final corpus-wide alignment. By using a majority vote, the final alignments can produce additional consensus resources.

2.1 Document-level alignment

To create document-level alignments, documents from each corpus are processed sequentially. Suppose $L = \{D_1, D_2, ..., D_k\}$ is a set of parallel documents where $d_{i,j}$ is the jth token in D_i. A matching $\mathcal{M}_L$ is a document-level alignment which consists of a set of relations, $R_1, ..., R_r$. For example, if $|L| = 2$, then $\mathcal{M}_L$ is a one-to-one word alignment between two lines of text. Its relations are either an aligned pair of tokens or an unaligned singleton token.

In Algorithm 1, $\mathcal{M}_L$ is generated by adding the next document to an existing matching and weighing the edges between tokens and relations according to Equation 1. Edges are pruned to both speed up the solver and avoid conflating separate or weakly related tokens. The maximum weighted matching is then used to decide which relations are expanded.

$$W(d_{i,j}, R_k) = \sum_{d \in R_k} f(d, d_{i,j}). \tag{1}$$

Algorithm 1 Document-level alignment

```
function AlignDocuments(L)
    $L = D_1, D_2, ..., D_k$                      ▷ Assume fixed ordering
    $\mathcal{M} = \{\{d_{1,j}\} : d_{1,j} \in D_1\}$
                         ▷ Initialize matching with singletons relations
    for $D_i = D_2, ..., D_k$ do
        $(V, E) = (D_i \cup \mathcal{M}, D_i \times \mathcal{M})$
        $G = (V, E, W(E))$               ▷ $W$ is defined by Eqn. 1
        $G' = \text{Prune}(G)$                  ▷ Remove small edges
        $A = \text{MaxWeightMatching}(G')$
        for $(d, R) \in A$ do
            $R \leftarrow R \cup \{d\}$
        $\mathcal{M} \leftarrow \mathcal{M} \cup \{\{d_{i,j}\} : d_{i,j} \in D_i \setminus A\}$
            ▷ Update existing relations or create new singletons
    return $\mathcal{M}$
```

The scoring function, f, is a weighted sum of the features described in Table 1. While a feasible weight function could be normalized by $|R_k|$, we instead choose to sum. If two different relations have a similar meaning but are not initially placed together, both will grow as tokens from new tokens would have a similar score to both. By using total score, the bigger relation will dominate. On the other hand, taking the sum could lead large relations matching with unrelated tokens. For ease of future analysis, errors of this type were preferred.

Feature	Values	Description
IDENTITY	Binary	$d = e$
PAIRWISE	Binary	Aligned(d, e)
LEMMA	Binary	Lemma(d) = Lemma(e)
POS	Binary	POS(d) = POS(e)
PARENT	Binary	Parent(d) = Parent(e)
NEIGHBORS	Integer	$\|L^+(d) \cap L^+(e)\|$, $L^+(d)$ is the multiset of outgoing edge labels from d in the dependency parse.
CHILDREN	Integer	The number of children u, v of d, e where the edges (d, u) and (e, v) have the same label and Aligned(u, v).
PARENT(V)	Real	Relates a child v of e to d by considering the set U that aligns to v. For each $u \in U$, we increment the score if its parent is d and give additional points if the parent's POS tag and edge label is the same as those of e. However, these are normalized by $\|U\|$.

Table 1: These specific features are used in $f(d, e)$. Aligned(d, e) is determined by the bitext aligner, and PARENT(V) is a feature for each child v of e. All features have weight 1, except for IDENTITY, which has weight 3. The pruning threshold is 4. These values could be further tuned.

$$F(\mathcal{M}) = \sum_{R_i \in \mathcal{M}} \sum_{d,e \in R_i} f(d, e) \tag{2}$$

A matching is scored by summing pairwise scores in all of its relations (Equation 2). Ideally, we would directly maximize $F(\mathcal{M})$, but that is NP-hard.[3] Instead, we match each document greedily.

2.2 Creating a corpus alignment

Suppose the documents in the multi-parallel corpora $\mathcal{C} = C_1, C_2, \ldots, C_c$ are already aligned, so for any document, we can find its counterpart in each C_i, if it exists. This is the case for the Biblical data since the verse numbers act as document labels.

Given an existing document-level alignment, we can improve the accuracy of individual features. For example, tokens within the same relation are synonymous, and so they are used to recompute the PAIRWISE feature. Algorithm 1 depends on the initial ordering of L. This motivates Algorithm 2,

[3] With just three documents, this is a weighted variant of the 3-dimensional matching problem.

which both recomputes the feature values and shuffles the documents between each of the $T = 10$ iterations.

Algorithm 2 Corpus alignment

```
**Input:** Multi-parallel corpora $\mathcal{C} = C_1, C_2, ...C_c$
Parse and tag each $C_i \in \mathcal{C}$ [4]
**for** $t = 1 \dots T$ **do**
    Recompute corpus statistics
    Align every pair $C_i, C_j \in \mathcal{C}$
    $\mathcal{C}$.shuffle()    ▷ Choose a new order for the documents
    **for all** documents $L \in \mathcal{C}$ **do**
        $\mathcal{M}_{L,t} = \textsc{AlignDocuments}(L)$
**Output:** $\{\text{argmax}_{\mathcal{M}_{L,t}} F(\mathcal{M}_{L,t}) : L \in \mathcal{C}\}$
```

2.3 Dataset and tools

The corpus of Bibles were collected by Mayer and Cysouw (2014) and contains 27 English versions. 23 contain just the New Testament (~8K verses, ~200K words), while four also include the Old Testament (~31K verses, ~900K words).

We use `fast_align` (Dyer et al., 2013), a greedy, transition-based dependency parser (Honnibal and Johnson, 2015), and an averaged perceptron POS tagger with Brown cluster features (Collins, 2002; Koo et al., 2008) for feature computation.[5]

3 Results

3.1 Analysis of alignments

Fig. 3 shows an example of an alignment produced by our system.[6] There are a few misalignments due to both literary divergence and our design choices, such as the sum in Equation 1 and targeting one-to-one alignments.

The visualization of the matchings simplifies analysis of literary variation. Relations with only a few or different members are easy to spot. These anomalies can be indicators of different choices in translation, unclean source text, or use of older language. For example, unlike “wise men" and “magi," “astrologers" only appears once across the rows **wise men** in Fig. 3, so the word choice may be a deliberate. Small or singleton relations prompt further investigation into the source.[7] Since some of our features are independent of meaning, we correctly align non-English (e.g. Hebrew) words.

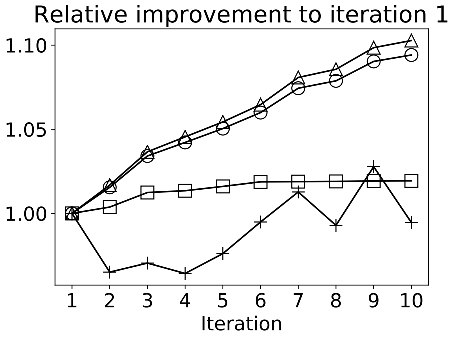


Figure 2: Relative change in total score with respect to the first iteration: Entire text (○); Old Testament (□); New Testament (△); per iteration (+). Since the feature weights are fixed, the absolute score is not meaningful.

Word indices in each of the columns also show the degree of reordering, which itself is a measure of divergence.

3.2 Improvements across iterations

Fig. 2 tracks the relative change in total score summed across all the documents. The Old Testament plateaus early, possibly because there are only four sources. The high variance in per iteration score shows the large effect of the ordering.

3.3 Limitations

Since there are no gold-standard annotations for this task, it is difficult to perform a meaningful quantitative evaluation on the alignments directly. Empirical evaluation is also challenging due to the scale of even a single multiway alignment.

Because the tools used are not trained specifically for historical English religious texts, it is possible for the features themselves to be imprecise or noisy. For this predominantly modern English corpora, tokens with incorrectly preprocessed features can still be placed in the correct relation due to the correct features. For an even better alignment, additional features or a looser definition of IDENTITY would encourage improved alignments.

The proposed method assumes a one-to-one mapping between tokens and relations. Multi-word expressions in the consensus, like **wise men**, can be viewed as a multi-relation expression. It is challenging to generalize our method to many-to-many alignments.

4 Additional Biblical Resources

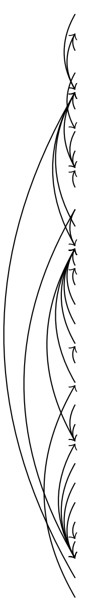
For a given relation, a consensus annotation can be derived by taking the majority annotation of the

[4] If parsing or tagging is improved by the alignments from the previous iteration, it could be rerun every iteration.

[5] Implemented by `https//spacy.io`.

[6] More examples in Appendix Tables 7 and 8

[7] In Matthew 8:22 (diaglot version), an error in the source was discovered by an extraneous singleton relation: {“The”}.



Parse	Consensus Analysis		worldwide	newsimplified	montgomery	etheridge	godsword	majority	lexham	common	contemporary
	Then	RB	(Then, 0)	(Then, 0)	(Thereupon, 0)	(Then, 0)	(Then, 0)	(Then, 0)	(Then, 0)	(Then, 0)	(first, 13)
	Herod	NNP	(Herod, 1)	(Herod, 1)	(Herod, 1)	(Herodes, 1)	(Herod, 1)	(Herod, 1)	(Herod, 1)	(Herod, 1)	(Herod, 0)
	,	.						(,, 2)			
	secretly	RB	(secretly, 10)	(secretly, 2)	(secretly, 3)	(privately, 2)	(secretly, 2)	(secretly, 3)	(secretly, 2)	(secretly, 2)	(secretly, 1)
	called	VBD	(called, 2)	(called, 3)	(sent, 2)	(called, 3)	(called, 3)	(called, 5)	(summoned, 3)	(called, 3)	(called, 2)
	for	IN			(for, 4)					(for, 4)	
	the	DT	(the, 3)	(the, 4)	(the, 5)	(the, 4)	(the, 4)	(the, 6)	(the, 4)	(the, 5)	(the, 4)
	wise	JJ	(wise, 4)				(wise, 5)	(wise, 7)	(wise, 5)		(wise, 5)
	men	NNS	(men, 5)	(astrologers, 5)	(Magi, 6)	(Magians, 5)	(men, 6)	(men, 8)	(men, 6)	(magi, 6)	(men, 6)
	,	.			(,, 7)	(,, 6)		(,, 9)			
	to	IN	(to, 6)	(to, 6)		*(to, 17)*					
	and	CC			(and, 8)	(and, 7)	(and, 7)		(and, 7)	(and, 7)	(and, 7)
	found	VBD	(talk, 7)	(find, 7)	(found, 9)	(learned, 8)	(found, 8)	(ascertained, 10)	(determined, 8)	(found, 8)	(asked, 8)
	of	IN	*(with, 8)*			*(at, 11)*					
	them	PRP	(them, 9)								
	He	PRP	(He, 12)								
	out	RP	(out, 14)	(out, 8)	(out, 10)		(out, 9)			(out, 9)	
	from	IN	(from, 15)		(from, 11)	(from, 9)	(from, 10)	(from, 11)	(from, 10)	(from, 10)	(in, 3)
	them	PRP	(them, 16)		(them, 12)	(them, 10)	(them, 11)	(them, 12)	(them, 11)	(them, 11)	(them, 9)
	exactly	RB	*(found, 13)*	(exactly, 9)			(exactly, 12)		(precisely, 9)	*(first, 18)*	
	the	DT		(the, 10)	(the, 13)			(the, 13)	(the, 12)	(the, 12)	
	exact	JJ*	(what, 17)			(what, 12)		(that, 15)			
	time	NN	(time, 18)	(time, 11)	(time, 14)	(time, 13)		(time, 14)	(time, 13)	(time, 13)	
	they	PRP	(they, 19)			(them, 18)					(they, 11)
	had	VBD					(had, 16)	(having, 4)		(had, 17)	(had, 12)
	when	WRB			(when, 15)		(when, 13)		(when, 14)	(when, 14)	(when, 10)
	the	DT	(the, 21)	(the, 12)	(the, 16)	(the, 14)	(the, 14)	(the, 16)	(the, 15)	(the, 15)	(the, 15)
	star	NN	(star, 22)	(star, 13)	(star, 17)	(star, 15)	(star, 15)	(star, 17)	(star, 16)	(star, 16)	(star, 16)
	appeared	VBN*	(saw, 20)	(appeared, 14)	(appeared, 18)	(appeared, 16)	(appeared, 17)	(appeared, 18)	(appeared, 17)	(appeared, 19)	(seen, 14)
	.	.	(., 11)	(., 15)	(., 19)	(:, 19)	(., 18)	(., 19)	(., 18)	(., 20)	(., 17)
	.	.	(., 23)								

Figure 3: A matching for document $D =$ Matthew 2:7. For space reasons, nine versions are shown in the same order they were aligned. Each row is a relation; the row header is its most common word. Each cell (r, c) is a member of the relation R_r and contains a token and an index from document D_c. The relations are arranged by the consensus index. Consensus POS tags and edges to the head token are shown in the analysis column. For presentation purposes, row headers are **bolded** if the majority of the documents in this table had a word in the relation. * indicates that there existed an equally competitive tag. Misalignments are *italicized*.

tokens in that relation. This can be extended on the corpus-level to word types by considering all relations represented by or containing that type.

4.1 In-domain paraphrases

We create a Bible-specific set of paraphrases. To obtain a distribution of similar words to a specific type w, we consider either all tokens in all relations that contain w (finer) or just the majority type of relations containing w (coarser) (Fig. 4).

Fine-grained paraphrases:
HEROD: `Herod` (0.90), `Herodes` (0.04), `he` (0.01) . . .
SECRET: `secret` (0.59), `mystery` (0.19), `private` (0.04) . . .
Coarse-grained paraphrases:
HEROD: `Herod` (0.95), `Herods` (0.05)
SECRET: `secret` (1.00)

Figure 4: Each word is followed by possible paraphrases and their proportions, which are computed from the consensus.

The domain specific paraphrases demonstrate the linguistic variation across the Bible, which can be further analyzed. Fig. 5 explores some of the variations that occur in our specific domain.

4.2 Consensus distributions

We can compute both the majority values (Fig. 3) and the entire distributions (Fig. 6) of specific features such as POS tags and head words. Aggregating each corpus independently before alignment, we can compute the possible tags for a word type. By using the alignments, the distribution of tags is softer. This could be useful as a prior in cross-lingual tag projection, since bitexts in the Bible are often not exact translations.

HYMENAEUS: `Hymenaeus` (0.82), `Hymenius` (0.04), `Hymeneus` (0.04), `Humenaios` (0.04) . . .
BLAZES: `burns` (0.48), `burning` (0.33) `burneth` (0.10), `blazes` (0.04) . . .
CHALLENGED: `said` (0.15), `opposed` (0.13), `urged` (0.12), `tested` (0.10), `tempted` (0.06), `tried` (0.05), `asked` (0.04).

Figure 5: These examples demonstrate spelling variation, language modernization, and unexpected domain-specific distributions.

For each possible head of a token, we compute consensus edge labels. By taking the most frequent edges, this results in a consensus dependency parse. If the proportions are used instead of the consensus, the result is a distribution over possible parses.

5 Discussion and Related Work

While monolingual insights like paraphrases have potential applications in semantic textual similarity (Agirre et al., 2012), there exist bigger corpora for those tasks, such as PPDB (Ganitkevitch et al., 2013). However, as the Bible is often the only significant parallel text for many of the world's languages, improved 27-way consensus English

POS tags
Before corpus alignments:
TIME: NN (1.00)
SECRET: NN (0.54), JJ (0.46)
With corpus alignments:
TIME: NN (0.94), NNS (0.05) . . .
SECRET: JJ (0.51), NN (0.47), NNS (0.01) . . .

Head words
Before corpus alignments:
TIME: `at` (0.23), `for` (0.09), `in` (0.07), `is` (0.06) . . .
SECRET: `in` (0.28), `is` (0.06) `kept` (0.04) `places` (0.04) . . .
With corpus alignments:
TIME: `at` (0.17), `in` (0.09), `for` (0.09), `is` (0.05) . . .
SECRET: `in` (0.32), `place` (0.05), `mystery` (0.04), `is` (0.04) . . .

Figure 6: A comparison of the POS tags (above) and head words (below) distributions for *time* and *secret* with and without the consensus alignment.

resources created here have value for annotation projection to low-resource languages.

The Bible has been productively used as a key resource for cross-lingual knowledge transfer (Yarowsky et al., 2001; Agić et al., 2015). Specifically, Johannsen et al. (2016) suggests a method for projecting POS tags and dependency parses onto a target language. Our approach can be modified in a similar way. By restricting the scoring function to use entirely language-independent features (e.g. pairwise alignments), our algorithm still maximizes the score of the matching by relearning an improved dictionary between iterations. The corpus alignment may also be desirable over separate alignments for multi-source projection tasks in noisier data because a word or phrase may only align with only a subset of the sources.

By generating resources specifically for the Bible, we hope to foster future computational methods for studying religious texts. Current Biblical visualization (Zhang et al., 2016) and authorship (Moritz et al., 2016) works use a small subset of the translations to perform their analysis. Our resources would encourage analysis across all versions of the Bible, which would be less biased than picking a small set. By weighing the votes cast by each token in a relation, it is even possible to emphasize a specific corpus.

The algorithms described in Section 2 can be applied to any parallel corpora. The scoring function is simple and accommodates arbitrary features. While our approach specifically assumes the documents (verses) within the corpora are already aligned, knowing which documents are similar (e.g. through clustering) is sufficient – perhaps at the cost of quality – to align and generate the subsequent resources.

6 Conclusion

We present a method for analyzing noisy multi-parallel text on significant multi-parallel corpora: 27 versions of the English Bible. The algorithm maximizes a flexible heuristic scoring function, so it is language-independent and applicable to any multi-parallel corpora. We produce a corpus-wide word alignment and use its consensus to create additional in-domain resources.

Given the Bible's unique role as the primary or only significant bitext for many of the world's languages, the robustly induced consensus analyses and associated alignments offer particular value to annotation projection in low-resource languages. In addition, these results shed insight into the underlying semantics of very widely studied source texts via both consensus and divergence of their multiple distinct translations.

Acknowledgments

This material is based upon work supported in part by DARPA LORELEI. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

References

Željko Agić, Dirk Hovy, and Anders Søgaard. 2015. If all you have is a bit of the Bible: Learning POS taggers for truly low-resource languages. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 268–272, Beijing, China. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada. Association for Computational Linguistics.

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Conference of the North American Chapter* *of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 16–23.

Michael Collins. 2002. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 1–8.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, Atlanta, Georgia. Association for Computational Linguistics.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

Anders Johannsen, Željko Agić, and Anders Søgaard. 2016. Joint part-of-speech and dependency projection from multiple sources. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 561–566, Berlin, Germany. Association for Computational Linguistics.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of ACL-08: HLT*, pages 595–603, Columbus, Ohio. Association for Computational Linguistics.

Moshe Koppel, Moty Michaely, and Alex Tal. 2016. Reconstructing ancient literary texts from noisy manuscripts. In *Proceedings of the Fifth Workshop on Computational Linguistics for Literature*, pages 40–46.

Shankar Kumar and William Byrne. 2004. Minimum bayes-risk decoding for statistical machine translation. In *In Proceedings of the NAACL 2004*, pages 169–176.

John Lee. 2007. A computational model of text reuse in ancient literary texts. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 472–479.

Thomas Mayer and Michael Cysouw. 2014. Creating a massively parallel Bible corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*.

Maria Moritz, Andreas Wiederhold, Barbara Pavlek, Yuri Bizzoni, and Marco Büchler. 2016. Non-literal text reuse in historical texts: An approach to identify reuse transformations and its application to Bible reuse. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1849–1859, Austin, Texas. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318.

Philip Resnik, Mari Broman Olsen, and Mona Diab. 1999. The Bible as a parallel corpus: Annotating the 'Book of 2000 Tongues'. *Computers and the Humanities*, 33(1):129–153.

Wei Xu, Chris Callison-Burch, and Bill Dolan. 2015. Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 1–11, Denver, Colorado. Association for Computational Linguistics.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the First International Conference on Human Language Technology Research*, pages 1–8, San Diego. Association for Computational Linguistics.

Ken Zhang, Carlos Folgar, and Jess McCuan. 2016. Decoding the Bible. `https://quid.com/feed/decoding-the-bible`.