Machine Translation for Historical Research: A case study of Aramaic-Ancient Hebrew Translations

Shmuel Liebeskind [a*], ChayaLiebeskind [b], Dan Bouhnik [c]
[a] *Department of Data Mining, Jerusalem College of Technology, Jerusalem, Israel*
E-mail: israellieb@gmail.com
[b] *Department of Computer Science, Jerusalem College of Technology, Jerusalem, Israel*
E-mail: liebchaya@gmail.com
[c] *Department of Computer Science, Jerusalem College of Technology, Jerusalem, Israel*
E-mail: bouhnik@g.jct.ac.il

Abstract. In this article, we investigate Machine Translation (MT) in a cultural heritage domain for two primary purposes: evaluating the quality of ancient translations and preserving Aramaic (an endangered language) by the ability to translate it to another spoken language. First, we detail the construction of publicly available Biblical parallel Aramaic-Hebrew corpus based on two ancient (early 2nd - late 4th century) Hebrew–Aramaic translations: Targum Onkelus and Targum Jonathan. Then using the Statistical Machine Translation (SMT) approach, which significantly outperforms the Neural Machine Translation (NMT) in our use case, validate the excepted high quality of the translations. The trained model fails to translate Aramaic texts of other dialects. However, when we train the same SMT model on another Aramaic-Hebrew corpus of a different dialect (Zohar - 13th century) a very high translation score is achieved. We examine an additional important cultural heritage source of Aramaic texts, the Babylonian Talmud (early 3rd - late 5th century). Since we do not have parallel corpus of the Talmud, we use the model trained on the Bible corpus for translation. We performed an analysis of the results and suggest some potential promising future research.

Keywords: Bible Translation, Neural Machine Translation, Statistical Machine Translation, Low-resource languages, Aramaic-Hebrew

1. Introduction

Statistical Machine Translation (SMT) has been researched for several decades. In the SMT paradigm translations are generated using statistical models whose parameters are automatically estimated from a parallel corpus. SMT systems were constructed for many spoken language pairs. In this research, we analyze SMT for the Aramaic-Hebrew language pair. Whereas Hebrew is a spoken language, Aramaic is dying as a spoken language. Since Aramaic is the language of some important Jewish religious texts, such as the Talmud and the book of Zohar, the ability to translate Aramaic texts to Hebrew must be preserved.

Recently, a Hebrew-Aramaic dictionary was incorporated into two important information retrieval systems: (1) the Responsa Project[1], a large database of Jewish texts which encompasses many important Jewish sources representing a period of over three thousand years of heritage and tradition, and (2) the Daf Yomi portal[2], a portal dedicated to supplying free, multi-level study material and lists of resources for those studying Talmud according to the 7-year Daf Yomi cycle. Tens of thousands of Jews worldwide study in the Daf Yomi program in which each of the 2,711 pages of the Babylonian Talmud are covered

[1] https://www.responsa.co.il
[2] http://daf-yomi.com

in sequence (see https://en.wikipedia.org/wiki/Daf_Yomi). The Hebrew-Aramaic dictionaries of these systems are based on the manual Aramaic-Hebrew dictionary by Ezra Zion Melamed [1]. While dictionary-based translation independently looks up each word in a bilingual dictionary, SMT considers the surrounding context of the words and typically provides better performance. SMT was by far the most extensively studied Machine Translation (MT) method prior to the advent of Neural Machine Translation (NMT). However, in our case study, SMT outperformed MT.

This paper makes the following contributions to MT research:

1. The construction of Aramaic-Hebrew parallel corpus based on translation of the Bible. We used the Corpus Encoding Standard [2], conforming to the level 1 annotation guidelines. Thus, our corpus is consistent with the corpus of Christodouloupoulos and Steedman [3] which includes 100 additional languages.
2. The evaluation of historical translations. Using SMT tools, we assess the quality of the constructed Aramaic-Hebrew parallel corpus. The corpus includes two of the most ancient Bible translations: Targum Onkelos (early 2nd or 4th century [4]) and Targum Jonathan (late 4th century [5]). Both translations are a combination of a commentary and a translation and include supplemental material in the form of aggadic[3] paraphrase. Therefore, assuming that it is easier to learn literal translation, a comparison between the quality of these two translations reveal their differences from word-to-word translation.
3. The evaluation of SMT algorithms trained over our Aramaic-Hebrew parallel corpus on other Aramaic texts of the same genre. We conducted a manual evaluation on the Talmud texts and an automatic evaluation on the Zohar texts which were translated to Modern Hebrew.

Our research is a crucial step in preserving the Aramaic language and cultural heritage. Our findings promote the development of the first Aramaic-Hebrew SMT system. A context-dependent system expected to replace the existing dictionary-based systems.

The Sections of this paper is organized as follows: Section 1 is the Introduction, in Section 2, we describe related work on SMT, NMT, MT for ancient languages and previous word on Aramaic NLP. In Section 3, we describe available Aramaic-Hebrew parallel corpora and the constructed Bible corpus. Subsequently, in Section 4, we detail our evaluation setting and results, accompanied by in-depth analysis. Finally, in Section 5, we summarize the primary findings and suggests future research directions.

[3] Aggadah is a compendium of rabbinical texts that combine legends, historical anecdotes, spiritual exhortations and practical advice in diverse fields, from business to medicine.

2. Background

Machine translation (MT) is the computerized automatic translation from one natural language to another. In this section, we first describe the Statistical Machine Translation (SMT) approach we found useful for our case study. Then, we discuss the more recent Neural Machine Translation (NMT) approach. Finally, we refer to MT for ancient languages and detail previous work on Aramaic NLP.

2.1. Statistical Machine Translation

Statistical machine translation (SMT) is an MT approach characterized by the use of machine learning methods. SMT systems are trained on large quantities of parallel data from which the systems learn how to translate small segments. Parallel data is a sentence-aligned corpus in two different languages, where each sentence in one language is matched with its corresponding translated sentence in the other language. The systems are then able translate previously unseen sentences.

Over the past decades, SMT has become the focus of MT research. There are some excellent tutorial introductions to SMT. Examples are: Jurafsky and Martin [6], Knight [7, 8], Koehn [9], Lopez [10]. In the next section, we describe the SMT approach we used for automatic Aramaic-Hebrew language pair translation, along with some related notable works.

The state-of-the-art models in SMT are phrase-based models. In phrase-based models, the unit of translation is any contiguous sequence of words, which we call a phrase. Each phrase in the source language is nonempty and translates to exactly one nonempty phrase in the target language. This is done using a simple mechanism.

1. The source sentence is segmented into phrases.
2. Each phrase is translated.
3. The translated phrases are permuted into a final order.

A phrase table contains the set of rules which governs the translation process, simply a list of all source phrases and all of their translations weighted by a conditional translation probability. The phrase table is gleaned from the training data.

Since translation systems produce alignments between source and target sentences, the training data must contain examples of these alignment decisions. However, the data available for training translation systems only contains sentence pairs.

Generally, there are two solutions to the sentence alignment problem. The first is to treat the unseen alignments as a hidden input and apply unsupervised learning methods [11–13]. The second is to first solve an easier problem of alignment, word alignments, and then use supervised methods for learning. The major drawback of the first solution is that unsupervised learning of phrase-based models requires numerous approximations and tradeoffs. Therefore, the alternative solution of training a word-based system and inferring phrase alignments heuristically from these word alignment is more commonly applied.

2.1.1. Word Alignment

Word alignment refers to word-based translation models [14], in which correspondence units between phrases are individual words. The aim of the word alignment task is to find the word-to-word correspondences in a sentence pair ($F_1^J = f_1...f_J, E_1^I = e_1...e_I$) where the source and target sentences include I and J words, respectively.

Och et al. [15] defined an alignment A of the two correspondences as:

$$A \subseteq \{(j,i) : j = 1,\ldots,J; i = 0,\ldots,I\} \tag{1}$$

in case that i = 0 in some (j, i) ∈ A, it means that the source word j aligns to an "empty" target word e_0.

In statistical world alignment models, the probability of a source sentence given target sentence is described as:

$$P(f_1^J|e_1^i) = \sum_{a_1^J} P(f_1^J, a_1^J|e_1^i) \tag{2}$$

where a_1^J indicates the alignment of the sentence pair. A few various parametric variants of $P(f_1^J, a_1^J|e_1^i) = p_\theta(f_1^J, a_1^J|e_1^i)$ have been suggested, and parameters of θ can be estimated using Maximum Likelihood Estimation (MLE) on a training corpus [15].

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \prod_{s=1}^{S} \sum_{a} p_\theta(f_s, a|e_s) \tag{3}$$

Viterbi alignment is the optimal alignment for the sentences.

$$\hat{a}_1^j = \underset{a_1^J}{\operatorname{argmax}} p_\theta(f_1^J, a_1^J|e_1^i) \tag{4}$$

The most common word alignment models are the IBM Models [14]. The IBM Models are a collection of word alignment models with growing complexity, beginning with probabilities of lexical translation, incorporating reordering and word duplication models. In combination with the Hidden Markov Model (HMM) [16], the IBM Models serve as a basis for most modern SMT systems.

The one-to-many mapping is one of the serious disadvantages of the IBM models. Their alignment function cannot return multiple values for a single input (many-to-one). To overcome this and enable many-to-many mappings, different methods are used for symmetrizing IBM-directed statistical alignment models.

2.1.2. Symmetrization

Och and Ney [15] introduced and compared various methods for computing word alignments using statistical models and two heuristic models based on the Dice coefficient. They presented various approaches for integrating word alignments to symmetrize directed statistical alignment models. Och and Ney [15] suggested an alignment model that was trained in both translation directions and obtained two alignments. They explored the space between intersection and union with expansion heuristics, which start with the intersection and incorporate additional alignment points. They found that a high-precision alignment of high-confidence alignment points could be achieved by intersecting the two alignments. If we choose the union of the two alignments, with additional alignment points, we achieve a high-recall alignment. A higher recall is more important in SMT [17], so an alignment union is likely to be selected [18]. Och and Ney [15] refined alignment models of first-order dependency and a fertility model show substantially better results than basic heuristic models. We used Och and Ney [15] symmetrization methods in our experiments, known also as *grow-diag-x* heuristics.

Venugopal et al., [19] presented a method of phrase extraction from alignment data generated by IBM Models. Their method creates phrase level knowledge sources that effectively represent local as well as global phrasal context by defining two kinds of scores within sentence consistency and across sentence consistency from the alignment. Their method also includes a phrase length scoring component based on the ratio of sentence length in the source and the target languages.

Zhang et al., [20] suggested an algorithm that simultaneously segments the sentences into phrases and finds their alignments. Sentence pair is represented by a two-dimensional matrix where the value of each cell corresponds to the point-wise mutual information (MI) between the source and target words. Then, the aligned phrase pairs are identified based on the similarities of MI values among cells. The joint probabilities of the identified phrase pairs is estimated using monolingual bigram language models. Eventually, the estimated joint probability distributions are marginalized into conditional probability distributions used by the decoder. They showed that their algorithm's precision is higher than the precision of a baseline system where phrase translations are extracted from the HMM word alignment.

Vogel et al., [21] presented a system that experiments four different approaches to phrase pair extraction: 1) harvesting the Viterbi path generated by a word alignment model, 2) adding additional alignment restrictions over the IBM-style alignment models [22], 3) using occurrence statistics collected over the entire corpus to achieve higher precision [19], and 4) integrating segmentation and phrase alignment [20]. They also added a generalization power by allowing for overlapping phrases to their system [23]. Their experimental results demonstrate that each phrase alignment approach yields different results when used alone, but combining the methods always improves the results.

While many models [15, 24–30] were only able to optimize the performance of the one-to-many model and the many-to-one model separately, and the predictions of the two models required combination with symmetrization heuristics, Fraser and Marcu [31] extended the IBM Models to produce symmetric many-to-many alignments that can be viewed as phrase alignments. They show that their generative alignment model is effective when trained using both unsupervised and semi-supervised training methods.

2.1.3. Decoding

Given a phrase table, it is possible to translate new input sentences. This process is called decoding. In SMT, decoding corresponds solving the following maximization problem [32]:

$$\hat{e}_1^I = \underset{e_1^I}{\operatorname{argmax}}\{P(e_1^I|f_1^J)\} \tag{5}$$

or after applying Bayes theorem:

$$= \underset{e_1^I}{\operatorname{argmax}}\{P(e_1^I) \cdot P(f_1^J|e_1^I)\} \tag{6}$$

Now, we have two models:

1. The language model $P(e_1^I)$
2. The translation model $P(f_1^J|e_1^I)$, the learned phrase table

Due to the constraints of the translation model, the number of possible outputs is finite. However, there is still a very large number of possible outputs to consider in order to maximize the function. Therefore, a primary objective of decoding is to search this space as efficiently as possible.

Many of the decoding algorithms follow a general framework detailed by Wang and Waibel [33] and Koehn [34], which is similar to the one used by Jelinek [35] for speech recognition. In this framework, search proceeds through a directed acyclic graph of states. Each state represents partial or completed hypotheses, constructed from left-to-right in the target language word order. Hypotheses in this space are extended by adding one or more source word indices to the set of the positions of the source string that have been translated and appending one or more target words to the hypothesis string to produce a new state. The cost of a new state is the cost of the original state multiplied with the translation, distortion and language model costs of the added phrasal translation. Model-specific extension operators were implemented to apply this algorithm to IBM Model 4 [36, 37], phrase-based models [34], or other translation models [33, 38].

A risk-free optimization to reduce the search space is to recombine two hypotheses; if they agree in the source words covered so far, the last target words generated, and the end of the last source phrase covered. Only the cheaper hypothesis is kept [34, 39].

There are several approaches to solve the optimization in Equation 5. One approach is to use $A*$ heuris-

tics [39]. Another approach is to convert the optimization problem to linear integer programming [37]. A more common approach is to use *beam search* [24, 36, 37]. A beam search assumes that there are many acceptable translations of any sentence. Therefore, a certain amount of search error, when the decoder does not choose the optimal hypothesis, is permitted. The search space's hypotheses may be stored in one or more priority queues (often called stacks) and the search speed is optimized by pruning the search graph. There are two types of pruning, *threshold pruning*, where any hypothesis with a probability less than t times the probability of the best estimate in the same priority queue is removed, and *histogram pruning*, where only the n best hypotheses are kept in any priority queue (t or n is the size of the beam). A beam search can achieve large speedups with a very little loss of accuracy [24, 34, 36, 37].

2.2. Neural Machine Translation

Neural machine translation (NMT), a new approach to MT that uses a large artificial neural network to predict the likelihood of a sequence of words has recently been proposed, [40–45]. Unlike the phrase-based SMT which consists of many small sub-components that are tuned separately, NMT aims at building a single neural network that can be jointly tuned to maximize the translation performance. Many of the NMT methods implement the concept of a sequence-to-sequence network [41] in which two Recurrent Neural Networks (RNNs) working together to turn one sequence into another. The network encoder condenses the input sequence into a vector, and the network decoder unfolds the vector into a new sequence. In many primary tutorials (such as https://sites.google.com/site/acl16nmt/home), NMT was widely described.

Transformer-based models have contributed to a massive improvement in the accuracy of neural machine translation in the past few years. The core feature of the transformer architecture, which follows the encoder-decoder paradigms, is the so-called multi-head attention mechanism [46], which enables the model to concurrently concentrate on various parts of the input. First, attention mechanisms were used in combination with Recurrent Neural Networks (RNNs), between the encoder and decoder. Then, the Transformer architecture replaced the RNNs in the encoder and decoder by multiple attention heads. The Transformer architecture is now the state-of-the-art architecture for NMT. We note rather extensive reviews of NMT are available [47–49], and refer the reader interested in more details to those works.

The performance of NMT declines sharply under low-resource constraints, underperforms SMT and requires significant volumes of auxiliary data to provide competitive results [50]. The effectiveness of Transformer under low-resource conditions is highly dependent on the hyper-parameter settings. Araabi and Monz [51] showed that, compared to using the Transformer default settings, using an optimized Transformer for low-resource conditions improves the translation quality up to 7.3 BLEU points.

A particularly promising approach for employing resources from additional languages is transfer learning [50, 52]. Transfer learning is training a model (child) with a limited amount of parallel corpus with a pre-trained model (parent) as the initial parameters in the training. The theory is that the information maintained in the parent model should be transferred to the newly trained child model. The parent and child pairs might share the target language [53]. The pairs might be both low-resource languages either related [54], or unrelated [55].

Many recent studies of NMT [56–62] use OpenNMT [63–65], an open-source toolkit for NMT.

In this study, we focused on the SMT approach for two reasons. First, in our case study, the SMT method significantly outperformed the NMT method implemented using OpenNMT. Second, due to the modularity of the SMT approach, interpretation of the results is much easier using SMT.

2.3. Machine Translation for ancient languages

Researches on NMT for ancient languages are beginning to emerge. Some studies have applied NMT for translating various ancient languages to their modern counterpart, such as Korean [66], Japanese [67] and Greek [68]. While these works focused on tokenization, character recognition, and missing characters recovery using deep neural networks. In the present research, the given ancient text is complete and readable. Moreover, we did not translate an ancient language to its modern counterpart, but translated an ancient language (Aramaic) to another ancient language (Biblical Hebrew).

Resnik et al. [69] recognized the Bible as a source for various NLP applications, such as lexical semantics, translation lexicons development, and evaluation of semantic tagging for multilingual machine translation. They used 13 different Bible translations to cre-

ate a parallel corpus. Christodouloupoulos and Steedman [3] followed them and increased the number of languages to 100 (4,950 unique language pairs) using the same xml format. Their corpus is publicly available. Therefore, for consistency, for this research we used the same Corpus Encoding Standard [2]. A corpus of 847 Bibles was created by Mayer and Cysouw [70] and was extended to 1556 Bibles in 1169 [71]. However, this corpus is no longer available.

Recently, the Johns Hopkins University Bible Corpus (JHUBC) [72], a massively parallel corpus in more than 1600 languages was presented. The corpus contains more than 4000 unique translations of the Christian Bible and counting, including 7400 lines from the New Testament in Aramaic. However, the corpus is not publicly available due to copyright restrictions. Additional diachronic and synchronic parallel corpus of Bible translations, the EDGeS corpus[73], was recently presented. It includes Bible translations in Dutch, English, German and Swedish, with texts dating back to the 14th century. The entire corpus is accessible to researchers by the well-known OPUS search infrastructure and the open portion of the corpus is available for download.

The Bible corpus is commonly used for multilingual studies [74, 75]. However, it is not sufficient for training a general-purpose MT system and is often used in combination with other parallel corpora, especially in the case of low-resource languages [76, 77].

2.4. Aramaic NLP

Several studies were conducted on Aramaic NLP. Few works used the Responsa project as a corpus of old texts in mixed Hebrew and Aramaic. These studies have addressed various tasks such as abbreviation disambiguation [78, 79], citations identification [80, 81], temporal data mining [82, 83], and diachronic thesaurus construction [84–87], etc. These experiments offered insights into the Aramaic language. But, since the Responsa project's primary language is Hebrew, these experiments did not concentrate explicitly on the Aramaic NLP.

Snyder and Barzilay [88] proposed a non-parametric model that jointly segments and identifies cross-lingual morpheme patterns for morphological segmentation of multiple languages. Priory, for each language and for the cross-lingual links, the model uses Dirichlet process. The model, using a multilingual parallel corpus of short phrases from the Hebrew Bible and translations in four languages: Arabic, Hebrew, Aramaic, and English was implemented (for Aramaic, Targum Onkelos was used (see Section 3)). Snyder and Barzilay [88] illustrated that in monolingual models the joint model dropped error was up to 24%. The model obtained better results in languages of the same family. However, for the English and Aramaic portion of the corpus, Snyder and Barzilay [88] did not have gold standard segmentation and their assessment is limited to Hebrew and Arabic.

The latest study on the discovery of parallel passages in the Babylonian Talmud corpus by Shmidman et al. [89] is an exception. As opposed to the Responsa project, the key language of the Talmud is Aramaic. The Shmidman et al. [89] method enables shifts to occur on word and phrase levels between parallel passages. They concentrated on the essence of the words at word level. First, the method used the input corpus to measure the Hebrew letter frequency. The two most infrequent letters were then listed for each word and the word was represented by those two letters. The method essentially removed prefixes letters and matres lectionis because they are the most frequent letters in the language. At phrase level, both four-grams and non-contiguous skip-grams were compared. For each five-word-sequence of text, they retrieved all four-word combinations that might exclude each of the last four words. To verify a certain match, a two-dimensional graph was created to cluster the matching skip-grams. Every match of the skip-grams was drawn on one axis by the starting word position of the base skip-gram and on the other axis by the corresponding starting word position of the skip-gram. Cases where a cluster with more or less diagonal line matches several skip-grams were deemed valid. since it compiles its list of possible matches during a pre-processing single pass of the text, the Shmidman et al. [89] method is capable of processing text of any size in $O(N)$ time. We note that they did not discover parallel passages suitable for SMT, but identified rephrasing and orthographic variation of a particular passage.

Recently, Liebeskind and Liebeskind [90] proposed a technique for automatic construction of Aramaic-Hebrew translation Lexicon. First, using a state-of-the-art word alignment translation model, they created an initial translation lexicon. Then, using string similarity measures of three types, they filtered the original lexicon: similarity between terms in the target language, similarity between a source and a target term, and similarity between terms in the source language. In their tests, for each type of similarity, Liebeskind and Liebeskind [90] used a parallel corpora of Bib-

lical Aramaic-Hebrew sentence pairs and examined several string similarity measures. They demonstrated their technique's advantage and its impact on precision and F1. More precisely, they showed that their filtering method significantly exceeds a filtering method based on the probability scores generated by a state-of-the-art word alignment algorithm. In the current paper, we use the same corpora to develop a complete end-to-end Aramaic-Hebrew translation system.

3. Parallel Corpora

SMT requires parallel data for learning. In a sentence-level parallel corpus, for every sentence in the source language there is a translated sentence in the target language. We used three Aramaic-Hebrew corpora:

1. **Targum Onkelos**, the Jewish Aramaic Targum, is an original translated text of the Pentateuch (Five Books of Moses), which was thought to have been composed in the early 2nd century CE or in the late 4th-early 5th centuries [4]. Its authorship is historically credited to Onkelos, a well-known convert to Judaism in the Tannaic period (c. 35-120 CE). The Talmud (Megillah 3a) relates that Targum Onkelos's content was first passed to Moses at Mount Sinai by God, but it was lost and recorded by Onkelos afterward. The Aramaic translation of Onkelos is a literal word-by-word translation with very little supplemental content in the form of exegesis. However, where challenging biblical passages are found, Onkelos attempts to eliminate obscurities and ambiguities. Sometimes in his translated text he uses non-literally aggadical interpretations or extensions, typically where the original Hebrew is either an idiom, a homonym, or a metaphor.
2. **Targum Jonathan**, the formal translation of the eastern Aramaic to the Nevi'im (Prophets), the second major division of the Hebrew Bible. Its authorship is traditionally credited to Jonathan ben Uzziel, a follower of Hillel the Elder. The Talmud points out that the Targum Jonathan tradition derived from the last prophets "from the mouths of Haggai, Zechariah, and Malachi," (Megillah 3a). Its general style is like that of Targum Onkelos, originated in the land of Israel and accepted in Babylonia in the third century. The Babylonian Academies took Targum Jonathan to the Diaspora.
3. **Zohar**, the central work of Jewish spiritual literature, known as Kabbalah. The Zohar is a serious of books with a commentary on the spiritual elements and scriptural interpretations of the Pentateuch as well as mysticism, mythical cosmogony, and mystical psychology. The Zohar scriptural exegesis can be viewed an esoteric type of the rabbinical literature known as Midrash, which elaborates on the Pentateuch. The Zohar is primarily written in a vague, obscure Aramaic style. Some scholars claim that the Zohar Aramaic seems to be composed by someone whose mother tongue was not Aramaic and that words from Andalusi Romance and Galician Portuguese are included in the text. The Zohar was first established in Spain in the 13th century. The publication was made by Moses de León, a Jewish writer from 1240–1305. De León credited the work to Shimon bar Yochai ("the Rashbi"), a rabbinic sage active after the Siege of Jerusalem (70 CE) and the fall of the Second Temple during the Jewish-Roman wars. Some non-Jews delve into the Zohar out of curiosity, or in search of meaningful and practical answers about their lives and the purpose of creation and existence. The purpose of the Zohar is to help the Jewish people and infuse the Torah and the commandments with wisdom. The Hebrew translation of the Zohar is modern.

3.1. Aramaic-Hebrew parallel Bible corpus

We constructed an Aramaic-Hebrew parallel corpus based on translation of the Bible. For this purpose, we used two corpora: Targum Onkelus and Targum Jonathan. Thus, our corpus covers all the Hebrew Bible[4]. Although, there is Targum Jonathan for the Pentateuch section of the Bible, Targum Onkelus precedes Targum Jonathan and is more widely used, for example, the Jewish obligation of "Shnayim mikra veechad targum" involves reciting the Hebrew text of the weekly portion twice and then reciting Targum Onkelos once.

For consistency with the corpus of Christodouloupoulos and Steedman [3] which includes 100 additional languages, we used a well-defined universal format, the Corpus Encoding Standard (CES [2]),

[4]The Catholic Bibles and the Eastern/Greek Orthodox Bibles contain supplementary content from the Septuagint (texts translated into Koine Greek) and other sources in their Old Testaments.

conforming to the level 1 annotation guidelines. This means that in practice the Aramaic Bible is an XML file containing nested book and chapter-related elements and <seg> verse-related elements. The serial IDs were marked on each of the verses. Figure 3.1 shows two verses of Gen:1–2 in Aramaic.

Fig. 1. Two verses of Gen:1–2 in Aramaic from the Aramaic Bible XML file

```
<cesDoc>
<cesHeader>
...
<profileDesc>
    <langUsage>
        <language iso639="arc">
            Aramaic
        </language>
    </langUsage>
    <wsdUsage>
        <writingSystem id="utf-8"/>
    </wsdUsage>
</profileDesc>
</cesHeader>
<text>
<body id="Bible" lang="arc">
<div id="b.GEN" type="book">
<div id="b.GEN.1" type="chapter">
    <seg id="b.GEN.1.1" type="verse">בקדמין, ברא ה', ית שמיא, וית ארעא. </seg>
    <seg id="b.GEN.1.2" type="verse">וארעא, הות צדיא וריקניא, וחשוכא, על אפי תהומא
    ורוחא מן קודם ה', מנשבא על אפי מיא.</seg>
    ...
</div>
</div>
</body>
</text>
</cesDoc>
```

4. Evaluation

This section presents the evaluation of historical translations using SMT methods. We first assessed the quality of the constructed Aramaic-Hebrew parallel corpus (Section 4.1). Then, we evaluated the performance of the SMT algorithm, which was trained on our parallel corpus, on other Aramaic text of the same genre, but of different dialect or period.

4.1. Evaluation Setting

4.1.1. Datasets

For properly training the SMT algorithms, we divided each of our three corpora to three sets: train, validation, and test. Table 1 shows the number of parallel verses (in the case of the Zohar, paragraphs) in each of the sets. We note that for the Bible corpus (Targum Onkelus and Jonathan), we first unified the two corpora and then divided it to the three sets.

4.1.2. Evaluation measure

In our experiments, we compared the performance of the algorithms by a typically used MT measure: BLEU (Bilingual Evaluation Understudy). The BLEU scores for individual sentences are measured by comparing them with a set of high quality reference translations. Those scores are then averaged over the entire dataset to achieve an approximation of the overall quality of the translation. In our case, we simply counted matching unigrams in the candidate translation unigrams and the reference text. Intelligibility, word order or grammatical consistency are not taken into consideration.

Table 1

Number of parallel verses in each set of the corpora

Corpus	Train	Dev	Test
Targum Onkelus	5,000	746	100
Targum Jonathan	8,000	1,196	100
Targum Onkelus and Jonathan	13,000	2,042	100
Zohar	13,830	1,273	100

4.1.3. Machine translation algorithms

We experimented with two types of MT algorithms: statistical and neural. For SMT, we applied the phrase-based model, described in Section 2.1. We used the Moses statistical machine translation[5] [91], an open source toolkit that has been widely used in previous work, with its default settings. For NMT (Section 2.2), we applied two types of models: RNN and Transformer. We used the OpenNMT toolkit[6] [63, 64, 92] with its default settings. We also applied an additional variation of the Transformer, which was found useful in previous work [50], subword tokanization using the SentencePiece package[7].

4.2. Results

To evaluate the quality of the constructed Biblical Aramaic-Hebrew corpus, we used parallel corpus for MT. Table 2 shows the results of the MT algorithms on our corpus.

Table 2

The results of the MT algorithms on our corpus

Method	BLEU
SMT	44.25
RNN	28.58
Transformer	14.78
Transformer+subword units	11.76

[5] http://www.statmt.org/moses/
[6] https://opennmt.net/
[7] https://github.com/google/sentencepiece

The SMT approach significantly outperforms the NMT approach. When we used an NMT model with more parameters (i.e. the transformer models), the performance decreased. Since there is a huge gap between the SMT and the NMT BLEU scores, we did not make efforts to improve the NMT algorithms (as was previously done for low-resource languages, see Section 2.2). Instead, we adopt the SMT approach for the Aramaic-Hebrew translation.

The BLEU score of the SMT method is considered relatively high in MT. This may confirm the claim of historians that Targum Onkelus and Targum Jonathan are almost entirely a word-by-word literal translation of the ancient Hebrew. Moreover, Aramaic and Hebrew have the same alphabet and share some common words[8]. Translation from Hebrew to Aramaic using the same system also yielded a high BLUE score of 41.73.

When we ran our SMT system on the two corpora separately, the BLUE score of Targum Onkelus (50.11) outperformed the BLUE score of Targum Jonathan (37.02) significantly. This finding also conforms with the claim that the overall style of Targum Jonathan is very similar to that of Targum Onkelos, but appears to be a looser paraphrase of the biblical text [93].

To better understand the challenges of the Aramaic-Hebrew translation task, we analyzed the translation errors of the SMT method. We randomly sampled 30 sentences and analyzed unigrams in the candidate translation which were considered by the BLEU score as an incorrect match to the reference text. We found 95 incorrect matches which can be divided into five main types:

1. Wrong morphological form (29.47%): the word was correctly translated, but a wrong morphological form was chosen. For example, the Aramaic word זכותא (lit. charity) (zkwta)[9] was translated to צדקות (cdqwt) in plural, instead of צדקה (cdqh) in singular, the Aramaic word דעל (lit. come) (d`l) was translated to בוא (bwa) instead of its missing spelling בא (ba), and the Aramaic word מללנא (lit. speak) (mllna) was translated to אדבר (adbr) - 1st-person singular in future tense, instead of דברנו (dbrnw) 1st-person plural in the past tense.
2. Synonym (25.25%): the word was correctly translated to a different synonym, sometimes with a wrong morphological form (29.16% of the synonym cases). For example, the Aramaic word לאטעיותך (lit. tempt) (laT`iwtk) was translated to לפתתך (lpttk), instead of להדיחך (lhdixk), the Aramaic word מסכנא (lit. poor) (mskna) was translated to האביון (habiwn), instead of דל (dl), and the Aramaic word סברית (lit. hope) (sbrit) was translated to קויתי (qwiti), instead of פללתי (pllti).
3. Untranslated words (20%): these words were left in Aramaic and were not translated at all.
4. Completely wrong translation (13.68%)
5. Different number of words (9.47%): single words which were translated to two words in the reference text or vice versa. For example, the Aramaic phrase בת כן תתכניס לעמך (lit. then you will come to your people - then you will pass away) (bt kn ttknis l`mk) was correctly translated to אחרי כן תבוא לעמך (axry kn tbwa l`mk), but the reference text translated אחרי כן (axry kn) to a single word אחר (axr) and לעמך (l`mk) to two words אל עמיך (al `mik).

The rest of the cases (2.1%) were cases where the translation significantly changed the word order/syntax of the verses. Our analysis reveals that many of the cases that were considered as incorrect matches were actually reasonable translations with different word choice (types: 1, 2, and 5).

4.2.1. Adapting the model for other Aramaic texts

We evaluated the performance of the trained SMT model on the third available parallel corpus of the Zohar (see Section 3). In contrast to the Bible which was originally translated from ancient Hebrew to Aramaic, the Zohar was translated from a later Aramaic to Modern Hebrew. Due to the languages differences, the performance of the model decreased significantly and we obtained a BLEU score of 14.76.

In our efforts to explore the usefulness of the SMT for Aramaic-Hebrew translation, we trained the same model on the Zohar train set, thus overcoming the languages differences problem. We achieved a high BLEU score of 52.69; a much better result than that of the Bible parallel corpus (44.25).

Furthermore, we conducted a manual evaluation on the Talmud texts. The Talmud translation poses more challenges to SMT. First, there is no available parallel corpus of the Talmud for training. Second, the Talmud is an Hebrew-Aramaic corpus containing mixed-

[8] https://www.safa-ivrit.org/imported/aramaic.php

[9] To facilitate readability, we used a transliteration of Hebrew using Roman characters; the letters used, in Hebrew lexico-graphic order, are *abgdhwzxTiklmns`pcqršt*.

language sentences. Third, the Talmud Aramaic dialect (Jewish Babylonian Aramaic) is different from that of the Aramaic Bible. The bilinguality of the Talmud texts prevented us from randomly sampling the corpus. Hence, we asked domain experts to extract a list of Aramaic-only sentences for analysis. To increase the challenge, all those 30 sentences are idiom and paraphrases. For example, בהדי הוצא לקי כרבא (bhdi hwca lqi krba) (lit. because of the thorns the cabbage was damaged) - because of the wicked man his righteous neighbor was punished, and לא אסתייע מילתא (la astii` milta) (lit. it didn't come to pass) - it didn't work out.

Next, we analyzed the translation of those 30 sentences and suggest future research direction to improve the quality of the Talmud translation. The total number of analyzed words is 257, an average of 8.53 words per sentence. 110 of the words were unknown to the model (42.97%). Only 12 of the 146 (8.2%) translated words were incorrectly translated. About 5% of the words are shared by the two languages, for instance, אפילו (apilw) (lit. even) and זמר (zmr) (lit. song).

Since the system was trained on the Bible corpus, some of the sentence were translated in biblical terms. An interesting example is the phrase אם בכיר ולקיש כחדא יינץ (am bkir wlqiš kxda iinc) (lit. if the first grains and the late-ripening grains sprout at the same time) that was translated to אם יורה ומלקוש יחדיו יינץ (am iwrh wmlqwš ixdiw iinc) (lit. if the first rain and the last rain (at the end of the rainy season) sprout at the same time), where אם is a shared word and יינץ is unknown.

The large amount of unknown words caused many of the translation errors. The model had difficulty recognizing the words' context and incorrectly chose a translation for the word in a different meaning, usually its most frequent sense in the Bible. For example, the word חמרא (xmra) appears in three of our sentences. It has two senses in Aramaic: wine and donkey. The wine sense is more frequent in the Bible, thus in all the sentences חמרא was translated to wine. However, wine was a good choice only for one of our sentences. Another common reason for translation failure was translations which were based on a single appearance in the Bible. For example, ועיניה (w`inih) (lit. his eyes) was incorrectly translated to עין ימינו (`in iminw) (lit. his right eye).

The promising results of the SMT approach for Aramaic-Hebrew translation suggest that it will be possible to significantly improve the quality of the Talmud translation by constructing a suitable parallel corpus with a few thousands of sentences. Alternatively, we can leverage unrelated unaligned monolingual data to create additional training data that differs only marginally from the initial training data [94, 95]. Another approach that may improve the accuracy and coverage of SMT is to combine comparable data [96, 97]. For example, comparable corpus can be useful for adding induced translations of low frequency words [97].

5. Conclusions and Future Work

In this paper, we described the construction of a parallel Biblical corpus from two ancient Aramaic translations: Targum Onkelus and Targum Jonathan. The corpus was encoded using Corpus Encoding Standard and will be publicly available. We used the corpus to evaluate the quality of historical translation within the MT framework. Two MT approaches were investigated: SMT and NMT. Since the SMT approach significantly outperformed the NMT approach, SMT was used in our evaluation. Our findings confirmed the historians' assumption that the translations mainly consist of word-to-word translation and that Targum Onkelus is more accurate than Targum Jonathan.

We also applied the SMT model to other two Aramaic dialects. We observed the necessity to train the model on a dataset of the same dialect. In the case of the Zohar, where training data is available, the SMT approach achieved a very high BLEU score (52.69). In the case of the Talmud, where there is not any available training data, we performed a manual analysis of the SMT approach trained on the Bible corpus. We concluded that it is mandatory to use texts in the Talmud's dialect for developing a useful translation system.

In the future, for Aramaic-Hebrew translation, we plan to incorporate monolingual data or use comparable corpus to improve the SMT performance.

We also plan to use the presented methodology to explore the translation quality of other ancient language pairs, such as the Septuaginta Greek-Hebrew translation. We believe that NLP is useful for cultural heritage research and in particular for preserving endangered languages.

References

[1] E.Z. Melamed, *Aramaic-Hebrew-English Dictionary of the Babylonian Talmud*, Feldheim Publishers, 2005.

[2] N. Ide, Encoding Linguistic Corpora, in: *Sixth Workshop on Very Large Corpora*, 1998.

[3] C. Christodouloupoulos and M. Steedman, A massively parallel corpus: the bible in 100 languages, *Language resources and evaluation* **49**(2) (2015), 375–395.

[4] P.S. Alexander, "Targum, Targumim", *Anchor Bible Dictionary* **6** (1992), 320–31.

[5] B.P. Mortensen, *The priesthood in Targum Pseudo-Jonathan. Renewing the profession. Vol. 1*, Brill, 2006.

[6] D. Jurafsky and J.H. Martin, Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition, Prentice Hall, Pearson Education International, 2009, pp. 1–1024.

[7] K. Knight, Automating knowledge acquisition for machine translation, *AI Magazine* **18**(4) (1997), 81.

[8] K. Knight, A statistical MT tutorial workbook, in: *Prepared for the 1999 JHU Summer Workshop*, 1999.

[9] P. Koehn, *Statistical machine translation*, Cambridge University Press, 2009.

[10] A. Lopez, Statistical machine translation, *ACM Computing Surveys (CSUR)* **40**(3) (2008), 8.

[11] D. Marcu and W. Wong, A phrase-based, joint probability model for statistical machine translation, in: *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, Association for Computational Linguistics, 2002, pp. 133–139.

[12] J. DeNero, D. Gillick, J. Zhang and D. Klein, Why generative phrase models underperform surface heuristics, in: *Proceedings of the Workshop on Statistical Machine Translation*, Association for Computational Linguistics, 2006, pp. 31–38.

[13] A. Birch, C. Callison-Burch, M. Osborne and P. Koehn, Constraining the phrase-based, joint probability statistical translation model, in: *Proceedings of the workshop on statistical machine translation*, Association for Computational Linguistics, 2006, pp. 154–157.

[14] P.F. Brown, V.J.D. Pietra, S.A.D. Pietra and R.L. Mercer, The mathematics of statistical machine translation: Parameter estimation, *Computational linguistics* **19**(2) (1993), 263–311.

[15] F.J. Och and H. Ney, A systematic comparison of various statistical alignment models, *Computational linguistics* **29**(1) (2003), 19–51.

[16] S. Vogel, H. Ney and C. Tillmann, HMM-based word alignment in statistical translation, in: *Proceedings of the 16th conference on Computational linguistics-Volume 2*, Association for Computational Linguistics, 1996, pp. 836–841.

[17] F.J. Och, C. Tillmann and H. Ney, Improved alignment models for statistical machine translation, in: *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

[18] F.J. Och and H. Ney, A comparison of alignment models for statistical machine translation, in: *Proceedings of the 18th conference on Computational linguistics-Volume 2*, Association for Computational Linguistics, 2000, pp. 1086–1090.

[19] A. Venugopal, S. Vogel and A. Waibel, Effective phrase translation extraction from alignment models, in: *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, Association for Computational Linguistics, 2003, pp. 319–326.

[20] Y. Zhang, S. Vogel and A. Waibel, Integrated phrase segmentation and alignment algorithm for statistical machine translation, in: *Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003 International Conference on*, IEEE, 2003, pp. 567–573.

[21] S. Vogel, Y. Zhang, F. Huang, A. Tribble, A. Venugopal, B. Zhao and A. Waibel, The CMU statistical machine translation system, in: *Proceedings of MT Summit*, Vol. 9, 2003, pp. 54–61.

[22] D. Wu, Stochastic inversion transduction grammars and bilingual parsing of parallel corpora, *Computational linguistics* **23**(3) (1997), 377–403.

[23] A. Tribble, S. Vogel and A. Waibel, Overlapping phrase-level translation rules in an SMT engine, in: *Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003 International Conference on*, IEEE, 2003, pp. 574–579.

[24] R. Zens, E. Matusov and H. Ney, Improved word alignment using a symmetric lexicon model, in: *Proceedings of the 20th international conference on Computational Linguistics*, Association for Computational Linguistics, 2004, p. 36.

[25] A. Ittycheriah and S. Roukos, A maximum entropy word aligner for Arabic-English machine translation, in: *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2005, pp. 89–96.

[26] Y. Liu, Q. Liu and S. Lin, Log-linear models for word alignment, in: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, Association for Computational Linguistics, 2005, pp. 459–466.

[27] N.F. Ayan and B.J. Dorr, A maximum entropy approach to combining word alignments, in: *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, Association for Computational Linguistics, 2006, pp. 96–103.

[28] S. Lacoste-Julien, B. Taskar, D. Klein and M.I. Jordan, Word alignment via quadratic assignment, in: *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, Association for Computational Linguistics, 2006, pp. 112–119.

[29] P. Liang, B. Taskar and D. Klein, Alignment by agreement, in: *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, Association for Computational Linguistics, 2006, pp. 104–111.

[30] R.C. Moore, W.-t. Yih and A. Bode, Improved discriminative bilingual word alignment, in: *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2006, pp. 513–520.

[31] A. Fraser and D. Marcu, Getting the structure right for word alignment: LEAF, in: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007.

[32] R. Zens, F.J. Och and H. Ney, Phrase-based statistical machine translation, in: *Annual Conference on Artificial Intelligence*, Springer, 2002, pp. 18–32.

[33] Y.-Y. Wang and A. Waibel, Decoding algorithm in statistical machine translation, in: *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, 1997, pp. 366–372.

[34] P. Koehn, Pharaoh: a beam search decoder for phrase-based statistical machine translation models, in: *Conference of the Association for Machine Translation in the Americas*, Springer, 2004, pp. 115–124.

[35] F. Jelinek, *Statistical methods for speech recognition*, MIT press, 1997.

[36] C. Tillmann and H. Ney, Word reordering and a dynamic programming beam search algorithm for statistical machine translation, *Computational linguistics* **29**(1) (2003), 97–133.

[37] U. Germann, M. Jahr, K. Knight, D. Marcu and K. Yamada, Fast and optimal decoding for machine translation, *Artificial Intelligence* **154**(1–2) (2004), 127–143.

[38] S. Nießen, S. Vogel, H. Ney and C. Tillmann, A DP based search algorithm for statistical machine translation, in: *Proceedings of the 17th international conference on Computational linguistics-Volume 2*, Association for Computational Linguistics, 1998, pp. 960–967.

[39] F.J. Och and H. Ney, Statistical multi-source translation, in: *Proceedings of MT Summit*, Vol. 8, 2001, pp. 253–258.

[40] N. Kalchbrenner and P. Blunsom, Recurrent continuous translation models, in: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1700–1709.

[41] I. Sutskever, O. Vinyals and Q.V. Le, Sequence to sequence learning with neural networks, in: *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[42] K. Cho, B. van Merriënboer, D. Bahdanau and Y. Bengio, On the Properties of Neural Machine Translation: Encoder–Decoder Approaches, *Syntax, Semantics and Structure in Statistical Translation* (2014), 103.

[43] R. Sennrich, B. Haddow and A. Birch, Controlling politeness in neural machine translation via side constraints, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 35–40.

[44] M. Johnson, M. Schuster, Q.V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado et al., Google's multilingual neural machine translation system: Enabling zero-shot translation, *Transactions of the Association for Computational Linguistics* **5** (2017), 339–351.

[45] J. Lee, K. Cho and T. Hofmann, Fully character-level neural machine translation without explicit segmentation, *Transactions of the Association for Computational Linguistics* **5** (2017), 365–378.

[46] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser and I. Polosukhin, Attention is All you Need, in: *NIPS*, 2017.

[47] P. Basmatkar, H. Holani and S. Kaushal, Survey on Neural Machine Translation for multilingual translation system, in: *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, IEEE, 2019, pp. 443–448.

[48] R. Dabre, C. Chu and A. Kunchukuttan, A survey of multilingual neural machine translation, *ACM Computing Surveys (CSUR)* **53**(5) (2020), 1–38.

[49] S. Yang, Y. Wang and X. Chu, A survey of deep learning techniques for neural machine translation, *arXiv preprint arXiv:2002.07526* (2020).

[50] R. Sennrich and B. Zhang, Revisiting Low-Resource Neural Machine Translation: A Case Study, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 211–221.

[51] A. Araabi and C. Monz, Optimizing Transformer for Low-Resource Neural Machine Translation, in: *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 3429–3435.

[52] A.F. Aji, N. Bogoychev, K. Heafield and R. Sennrich, In Neural Machine Translation, What Does Transfer Learning Transfer?, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 7701–7710. doi:10.18653/v1/2020.acl-main.688. https://www.aclweb.org/anthology/2020.acl-main.688.

[53] B. Zoph, D. Yuret, J. May and K. Knight, Transfer Learning for Low-Resource Neural Machine Translation, in: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1568–1575.

[54] T.Q. Nguyen and D. Chiang, Transfer Learning across Low-Resource, Related Languages for Neural Machine Translation, in: *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2017, pp. 296–301.

[55] T. Kocmi and O. Bojar, Trivial Transfer Learning for Low-Resource Neural Machine Translation, in: *Proceedings of the Third Conference on Machine Translation: Research Papers*, 2018, pp. 244–252.

[56] D. Alvarez-Melis and T. Jaakkola, A causal framework for explaining the predictions of black-box sequence-to-sequence models, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 412–421.

[57] J. Ling and A.M. Rush, Coarse-to-Fine Attention Models for Document Summarization, *EMNLP 2017* (2017), 33.

[58] M.T. Ribeiro, S. Singh and C. Guestrin, Semantically equivalent adversarial rules for debugging nlp models, in: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 856–865.

[59] K. Murao, K. Kobayashi, H. Kobayashi, T. Yatsuka, T. Masuyama, T. Higurashi and Y. Tabuchi, A Case Study on Neural Headline Generation for Editing Support, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers)*, 2019, pp. 73–82.

[60] R.R. Riyadh and G. Kondrak, Joint Approach to Deromanization of Code-mixed Texts, in: *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, 2019, pp. 26–34.

[61] G. Klein, D. Zhang, C. Chouteau, J.M. Crego and J. Senellart, Efficient and High-Quality Neural Machine Translation with OpenNMT, in: *Proceedings of the Fourth Workshop on Neural Generation and Translation*, 2020, pp. 211–217.

[62] N. Ballier, N. Amari, L. Merat and J.-B. Yunès, The Learnability of the Annotated Input in NMT Replicating (Vanmassenhove and Way, 2018) with OpenNMT, in: *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 5631–5640.

[63] G. Klein, Y. Kim, Y. Deng, J. Senellart and A. Rush, OpenNMT: Open-Source Toolkit for Neural Machine Translation, in: *Proceedings of ACL 2017, System Demonstrations*, Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 67–72. https://www.aclweb.org/anthology/P17-4012.

[64] G. Klein, Y. Kim, Y. Deng, V. Nguyen, J. Senellart and A. Rush, OpenNMT: Neural Machine Translation Toolkit, in: *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, Association for Machine Translation in the Americas, Boston, MA, 2018, pp. 177–184. https://www.aclweb.org/anthology/W18-1817.

[65] G. Klein, F. Hernandez, V. Nguyen and J. Senellart, The OpenNMT neural machine translation toolkit: 2020 edition, in: *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (AMTA 2020)*, 2020, pp. 102–109.

[66] C. Park, C. Lee, Y. Yang and H. Lim, Ancient Korean Neural Machine Translation, *IEEE Access* **8** (2020), 116617–116625.

[67] T. Clanuwat, A. Lamb and A. Kitamoto, Kuronet: Pre-modern Japanese kuzushiji character recognition with deep learning, in: *2019 International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, 2019, pp. 607–614.

[68] Y. Assael, T. Sommerschield and J. Prag, Restoring ancient text using deep learning: a case study on Greek epigraphy, *arXiv preprint arXiv:1910.06262* (2019).

[69] P. Resnik, M.B. Olsen and M. Diab, The Bible as a parallel corpus: Annotating the 'Book of 2000 Tongues', *Computers and the Humanities* **33**(1) (1999), 129–153.

[70] T. Mayer and M. Cysouw, Creating a massively parallel Bible corpus, in: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 2014, pp. 3158–3163.

[71] E. Asgari and H. Schütze, Past, Present, Future: A Computational Investigation of the Typology of Tense in 1000 Languages, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 113–124.

[72] A.D. McCarthy, R. Wicks, D. Lewis, A. Mueller, W. Wu, O. Adams, G. Nicolai, M. Post and D. Yarowsky, The johns hopkins university bible corpus: 1600+ tongues for typological exploration, in: *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 2884–2892.

[73] G. Bouma, E. Coussé, T. Dijkstra and N. van der Sijs, The EDGeS Diachronic Bible Corpus, in: *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 5232–5239.

[74] C. Malaviya, G. Neubig and P. Littell, Learning Language Representations for Typology Prediction, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2529–2535.

[75] R. Östling and J. Tiedemann, Continuous multilinguality with language vectors, in: *15th Conference of the European Chapter of the Association for Computational Linguistics, Valencia, Spain, April 3-7, 2017*, Vol. 2, Association for Computational Linguistics, 2017, pp. 644–649.

[76] F. Guzmán, P.-J. Chen, M. Ott, J. Pino, G. Lample, P. Koehn, V. Chaudhary and M. Ranzato, The FLORES Evaluation Datasets for Low-Resource Machine Translation: Nepali–English and Sinhala–English, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 6100–6113.

[77] X. Wang, H. Pham, P. Yin and G. Neubig, A Tree-based Decoder for Neural Machine Translation, in: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 4772–4777.

[78] Y. HaCohen-Kerner, A. Kass and A. Peretz, HAADS: A Hebrew Aramaic abbreviation disambiguation system, *Journal of the American Society for Information Science and Technology* **61**(9) (2010), 1923–1932.

[79] Y. HaCohen-Kerner, A. Kass and A. Peretz, Initialism disambiguation: Man versus machine, *Journal of the American Society for Information Science and Technology* **64**(10) (2013), 2133–2148.

[80] Y. HaCohen-Kerner, N. Schweitzer and D. Mughaz, Automatically identifying citations in Hebrew-Aramaic documents, *Cybernetics and Systems: An International Journal* **42**(3) (2011), 180–197.

[81] M. Koppel and N. Schweitzer, Measuring direct and indirect authorial influence in historical corpora, *Journal of the Association for Information Science and Technology* **65**(10) (2014), 2138–2144.

[82] D. Mughaz, Y. HaCohen-Kerner and D. Gabbay, Mining and using key-words and key-phrases to identify the era of an anonymous text, in: *Transactions on Computational Collective Intelligence XXVI*, Springer, 2017, pp. 119–143.

[83] D. Moghaz, Y. Hacohen-Kerner and D. Gabbay, Text Mining for Evaluating Authors' Birth and Death Years, *ACM Transactions on Knowledge Discovery from Data (TKDD)* **13**(1) (2019), 7.

[84] C. Liebeskind, I. Dagan and J. Schler, Statistical thesaurus construction for a morphologically rich language, 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)*, 2012, pp. 59–64.

[85] H. Zohar, C. Liebeskind, J. Schler and I. Dagan, Automatic thesaurus construction for cross generation corpus, *Journal on Computing and Cultural Heritage (JOCCH)* **6**(1) (2013), 4.

[86] C. Liebeskind, I. Dagan and J. Schler, Semiautomatic construction of cross-period thesaurus, *Journal on Computing and Cultural Heritage (JOCCH)* **9**(4) (2016), 22.

[87] C. Liebeskind, I. Dagan and J. Schler, An Algorithmic Scheme for Statistical Thesaurus Construction in a Morphologically Rich Language, *Applied Artificial Intelligence* **33**(6) (2019), 483–496.

[88] B. Snyder and R. Barzilay, Unsupervised Multilingual Learning for Morphological Segmentation, in: *Proceedings of ACL-08: HLT*, Association for Computational Linguistics, Columbus, Ohio, 2008, pp. 737–745. https://www.aclweb.org/anthology/P08-1084.

[89] A. Shmidman, M. Koppel and E. Porat, Identification of Parallel Passages Across a Large Hebrew/Aramaic Corpus, *Journal of Data Mining & Digital Humanities* (2018).

[90] C. Liebeskind and S. Liebeskind, Automatic Construction of Aramaic-Hebrew Translation Lexicon, in: *Proceedings of LT4HALA 2020-1st Workshop on Language Technologies for Historical and Ancient Languages*, 2020, pp. 10–16.

[91] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens et al., Moses: Open source toolkit for statistical machine translation, in: *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, 2007, pp. 177–180.

[92] G. Klein, F. Hernandez, V. Nguyen and J. Senellart, The OpenNMT neural machine translation toolkit: 2020 edition, in: *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (AMTA 2020)*, 2020, pp. 102–109.

[93] A. Shinan, Dating Targum Pseudo-Jonathan: Some More Comments, *Journal of Jewish studies* **41**(1) (1990), 57–61.

[94] W. Yang and Y. Lepage, Inflating a training corpus for SMT by using unrelated unaligned monolingual data, in: *International Conference on Natural Language Processing*, Springer, 2014, pp. 236–248.

[95] K. Wołk, E. Rejmund and K. Marasek, Multi-domain machine translation enhancements by parallel data extraction from comparable corpora, *arXiv preprint arXiv:1603.06785* (2016).

[96] I. Skadiņa, A. Aker, N. Mastropavlos, F. Su, D. Tufis, M. Verlic, A. Vasiljevs, B. Babych, P. Clough, R. Gaizauskas et al., Collecting and using comparable corpora for statistical machine translation, in: *Proceedings of the 8th international conference on language resources and evaluation (LREC), Istanbul, Turkey*, 2012.

[97] A. Irvine and C. Callison-Burch, Combining bilingual and comparable corpora for low resource machine translation, in: *Proceedings of the eighth workshop on statistical machine translation*, 2013, pp. 262–270.