Ben-Gurion University of the Negev
The Faculty of Natural Sciences
The Department of Computer Science

Building a Hebrew FrameNet Lexical Resource from Parallel Movie Subtitles

Ben Eyal

Thesis submitted in partial fulfillment of the requirements
for the Master of Sciences degree

Under the supervision of Prof. Michael Elhadad

February 2018

אוניברסיטת בן־גוריון בנגב
הפקולטה למדעי הטבע
המחלקה למדעי המחשב

בניית משאב לקסיקלי ל־FrameNet בעברית
מתוך כתוביות מקבילות לסרטים

בן אייל

חיבור לשם קבלת התואר "מגיסטר" בפקולטה למדעי הטבע

בהנחיית פרופ' מיכאל אלחדד

פברואר 2018

Ben-Gurion University of the Negev
The Faculty of Natural Sciences
The Department of Computer Science

Building a Hebrew FrameNet Lexical Resource from Parallel Movie Subtitles

Ben Eyal

Thesis submitted in partial fulfillment of the requirements
for the Master of Sciences degree

Under the supervision of Prof. Michael Elhadad

Signature of student: ________________ Date: ________________
Signature of supervisor: ________________ Date: ________________
Signature of chairperson of the
committee for graduate studies: ________________ Date: ________________

February 2018

תקציר

אנו מציגים את גישתנו להעשרת המשאב הלקסיקלי של FrameNet ע"י ניצול זוגות של משפטים אנגלית־־עברית מתוך כתוביות מסרטים ותוכניות טלוויזיה. פיתחנו כלי חצי־־אוטומטי לוויזואליזציה ולסימון משפטים לפי איכותם, כאשר היעד הסופי הוא ללמוד מסווג לינארי מתוך כמות קטנה של דוגמאות שסומנו ידנית. בסוף נתאר כיצד המערכת שלנו משתלבת עם מערכת קיימת על בסיס FrameNet בעברית.

Abstract

We present our approach to enrich the Hebrew FrameNet lexical resource by exploiting English-Hebrew aligned sentence pairs of movie and TV subtitles. We have developed semi-automatic annotation and visualization tools with the goal of learning a linear classifier after a small number of manual annotations. In the end, we describe how our system can integrate with an existing Hebrew FrameNet framework.

Acknowledgements

First, I would like to thank my adviser, Prof. Michael Elhadad, for always finding time to help, wherever in the world he may be. I would also like to thank Avi Hayoun, Tal Baumel, Matan Eyal, Jumana Nassour-Kassis, Dan Schulman, Dr. Yael Netzer, and Dr. Meni Adler for always lending me their ears when I needed help. Thank you, Shahaf S. Shperberg, for all the coffee and YouTube videos, and last but not least, thank you Gali, my love, for all that you have done and for all you are doing. None of this would have been possible without you.

Contents

List of Figures

List of Tables

1 Introduction

Semantic role labeling (SRL) is a task that can best be described as annotating sentences with labels that answer questions such as “Who did what to whom, when, and where?”, and the answers to this question are called “roles”.

The two major SRL annotation schemes are FrameNet [4] and PropBank [34]. The work surrounding these schemes has contributed fully-annotated corpora in various languages, three CoNLL shared tasks, and many automatic systems.

Work on SRL have been shown to advance more complicated tasks such as question answering [21] and information extraction [12].

In this work we set out to contribute a new Hebrew corpus and lexical resources to further the advancement of the Hebrew FrameNet Project [19]. Chapter 2 further motivates the SRL task and its uses. Chapter 3 covers previous work relevant to our project. Chapter 4 describes our setup and the tools we chose to use for our task. Chapter 5 presents the research question and pinpoints our objectives. Chapter 6 elaborates on how the work was done and gives statistics on different sub-tasks. Finally, chapter 7 shows our final results and suggest future work that can be done to make them even better.

2 Motivation

2.1 Semantic Role Labeling Task

Formally, Semantic role labeling (SRL) is the task of automatically labeling predicates and arguments in a sentence with shallow semantic labels. Two primary applications of SRL are information extraction (IE) and question answering (QA) [12, 21].

Given the sentence "John broke the window" we can ask syntactical questions, *e.g.,* "Is 'broke' a verb?", "What is the head of the sentence?", and "How many noun phrases are in the sentence?". SRL tells us that "John" is the AGENT and "the window" is the THEME of a "breaking" event. This allows us to ask questions about the *meaning* of the sentence, *e.g.,* "What did John brake?" and "Who broke the window?". It is also possible to ask the same questions with a syntactically different sentence: "The window was broken by John", or "The window broke".

In order to perform SRL annotations on sentences, one needs to decide on the annotation guidelines: which labels are used to refer to each role (such as Agent or Theme in the example above). Two main approaches exist in linguistic theory to provide such an inventory of roles: PropBank [34] defines a set of generic labels which can be used for a wide range of events and relations; FrameNet [4] defines specific roles for each event type. Due to its size, PropBank (explained in more detail in §3.1.1) is currently the most widely used annotation scheme in works featuring automatic SRL. On the other hand, FrameNet provides annotations at a finer granularity which can be mapped easily to PropBank annotations.

Multilingual SRL

Multilingual SRL consists of performing the task of SRL on a variety of languages.

It is a theoretically and practically interesting extension of the task:

> **Theoretical.** Is the semantic frames repository stable across languages? or does it depend on each specific language?
>
> **Practical.** To enable QA and IE on a range of languages

Approaches in general build on the assumption that the same semantic frames inventory can be shared across languages - but this remains an empirical question to be verified that the same set of semantic primitives can be used across languages from different families.

Most approaches to multilingual SRL require English-to-target language parallel corpora. This poses no problem for languages rich with textual resources, but is very problematic with Hebrew, as there are barely any resources.

Akbik et al. [2] have shown that given the English PropBank and high-quality parallel corpora, one can generate PropBanks in other languages, albeit small. We investigate the same question in this work with respect to FrameNet in Hebrew.

2.2 Objectives

We aim to achieve two major objectives:

1. Develop a Hebrew FrameNet to support frame-semantic SRL, and by extension allow for complicated linguistic tasks in Hebrew such as QA and IE.

2. Show that a semi-automatic development of the lexical resource in Hebrew is possible using an English-Hebrew corpus and sentence alignments.

3 Previous Work

3.1 Semantic Role Labeling

3.1.1 Annotation Schemes for SRL

There are two main annotation schemes widely used when performing SRL: PropBank and FrameNet. As explained in Palmer et al. [34]:

> The PropBank project and the FrameNet project [...] share the goal of documenting the syntactic realization of arguments of the predicates of the general English lexicon by annotating a corpus with semantic roles.

FrameNet

FrameNet is an annotation scheme and a lexical database inspired by Fillmore's "frame semantics" [15]. Fillmore presented the concept of a 'semantic frame' as

> A system of concepts related in such a way that to understand any one of them you have to understand the whole structure in which it fits.

Baker et al. [4] introduced a new linguistic resource called FrameNet, on which work is ongoing to this day. The purpose of FrameNet is to realize the idea of frame semantics in English. This goal is being pursued by building a lexical database of annotated examples of how various words are used in actual texts, grouped by semantic frame.

In Ruppenhofer et al. [39] the project defines a formal structure for semantic frames, and various relationships between and within them.

The FrameNet lexical database is comprised of the following:

> **Frames.** Each frame contains a list of frame evoking words (such as "bought" in the sentence "John bought a car from Jane"). These are known as Frame Evoking Elements (FEEs) or Lexical Units (LUs). Additionally, each frame defines a list of participants and a list of constraints on and relationships between these participants. The participants are called Frame Elements (FEs).
>
> **Lexical Units.** Formally, an LU is a word lemma paired with a coarse part-of-speech tag and is unique within its frame. For example, both the words **bought** and **buying** are both represented by the LU **buy.v** in the COMMERCE_BUY frame.
>
> In the FrameNet formalism, LUs can have almost any part-of-speech tag. As an example consider the noun **purchase** (as in "a purchase was made"), which is one of the LUs in the COMMERCE_BUY frame. That being said, verbs are the most common LUs.
>
> A **target** is any constituent which evokes a frame. It is the instance of an LU in a given piece of text. For example, both **buying** and **bought** could be targets which evoke the COMMERCE_BUY frame and they represent the LU **buy.v**.
>
> **Frame Elements.** FrameNet classifies frame elements in terms of how central they are to the frame. The two primary levels of centrality are labeled "core" and "peripheral".
>
> An FE is classified as a *core* FE if it instantiates a conceptually necessary component of a frame, while making a frame unique. For example, in the COMMERCE_BUY frame, the FEs **Buyer** and **Goods** are considered core elements, while the FE **Money** (representing the thing given in exchange for the goods) is not.
>
> In determining which FEs are core in their frame, a few formal properties are considered which may provide evidence for core status:
>
> - When an element must be explicitly expressed, it is core. For instance, *resemble* requires two entities to compare.
> - If an element receives a definite interpretation when omitted, it is core. For example, the sentence "John arrived." is incomplete if the goal location at which John arrived cannot be inferred from context.

> FEs that do not introduce additional, independent or distinct events from the main reported event are classified as *peripheral*. Peripheral FEs mark notions such as time, place, manner, degree, etc. of the main event represented by the frame.

PropBank

Kingsbury and Palmer [23] presented a new semantic lexicon project called PropBank (Proposition Bank). The goal of the project was to add verbal predicate-argument structure information to the Penn English Treebank II corpus. They strove to define a labeling scheme which would be consistent across syntactic alterations. For example, in the sentences "The window broke" and "John broke the window", the window is the *Patient* argument of the verbal predicate *broke*.

PropBank annotations are closely tied to syntax, due to the dataset being based on phrase-structure syntax trees from the Penn Treebank II corpus.

PropBank defines six *core roles*, ARG0-ARG5, which receive different interpretations for different predicates.

Argument	Interpretations
ARG0	Agent
ARG1	Patient
ARG2	Instrument, Benefactive, Attribute
ARG3	Starting point, Benefactive, Attribute
ARG4	Ending Point

Table 3.1: List of arguments and some of their possible interpretations in PropBank as described in Bonial et al. [6]

Additional modifier role labels, with the form ARGM-*, are defined, such as ARGM-TMP (temporal role) and ARGM-DIR (directional role) which maintain their interpretations across predicates, and are general in their semantic meanings.

As a result, the PropBank formalism has a small number of roles, thus making the use of the resource as a training set attractive from the perspective of machine learning.

The PropBank project represents a semantic theory which postulates that most events which occur in natural language text are represented by verbs, and that the semantic roles which participate in these events are specific to each and every predicate. This would seem counter-intuitive given the small number of roles. The reason for this seeming contradiction is the conscious effort that was made to make the role labels as consistent across frame-sets as possible.

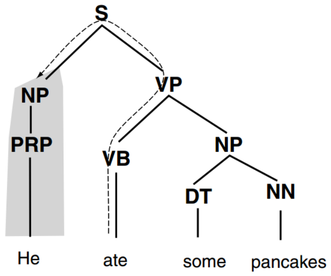
NomBank was an annotation project that is related to the PropBank project. Their goal was to mark the sets of argument which co-occur with nouns in the PropBank corpus, in a similar manner in which the PropBank project records such information for verbs. The project is described in further detail in Meyers et al. [28].

3.1.2 SRL Methods

Work on automatic SRL systems began with Gildea and Jurafsky [16], where they treat SRL as a sequence-tagging problem. They split the task into two sub-tasks: frame identification and role labeling, where the former is given, and work is done on the latter. Their approach was training a statistical classifier on roughly 50,000 FrameNet example annotated sentences, as opposed to full-text annotations, to get a probability distribution across all possible roles. The features used are:

> **Phrase Type.** One of PTB's syntactic categories, as *e.g.,* noun phrases and prepositional phrases are more likely than adverbial phrases to be frame elements.
>
> **Governing Category.** Either S for subjects of verbs, or VP for objects of verbs, *e.g.,* in Figure 3.1, the governing category of the constituent "some pancakes" is VP, and S for "He".
>
> **Parse Tree Path.** The path from the target word through the parse tree to the constituent in question. An example can be found in Figure 3.1.
>
> **Position.** Indicates whether the constituent to be labeled occurs before or after the predicate defining the semantic frame.
>
> **Voice.** Either active or passive.
>
> **Head Word.** The head words of noun phrases.

Figure 3.1: Example of the path feature for the constituent "He"



A separate model for role annotation was constructed for each frame. Ideally, each one would be created as follows:

$$P\left(r \mid h, pt, gov, position, voice, t\right) = \frac{\#\left(r, h, pt, gov, position, voice, t\right)}{\#\left(h, pt, gov, position, voice, t\right)}$$

Where r is the semantic role, h is the head word, pt is the phrase type, and t is the target word. However, it is hard to achieve such distributions, as the low number of training sentences per target word lead to a sparse training set. For this reason, the actual classifier was built by constructing a linear combination of distributions conditioned on various subsets of the features.

To evaluate the performance of the system on the role-labeling sub-task, three measures were defined: *Coverage*, which indicates the percentage of the test data for which the correct annotation had been seen in the training process; *Accuracy*, which is the proportion of the test data for which the correct label is deemed most likely by the model; and *Performance*, which is the product of the two.

Their system achieved 82% accuracy in the role-labeling task, and 65% precision and 61% recall on both sub-tasks.

Xue and Palmer [45] hypothesized that most SRL systems until the time of writing did not take full advantage of the input to the SRL task, a syntactic parse tree. They break the SRL task into three sub-tasks:

Pruning. Filter out constituents that are very unlikely to be arguments.

Argument Identification. Classify whether the candidates from the previous step are semantic arguments or not.

Argument Classification. Finally, run a multi-class classifier to classify candidates into one of the classes, or NULL.

Some of the new features proposed by Xue and Palmer are:

Lexicalized constituent type. A combination of the predicate lemma and the phrase type, *e.g., give_np*.

Lexicalized head word. A combination of the predicate lemma and the head word, *e.g., give_states*.

Voice position combination. A combination of voice and position features *e.g., passive_before*.

Xue and Palmer show comparable results to the state-of-the-art at the time, but use less features and a Maximum Entropy classifier that was faster to train, but shown to give worse results than a SVM.

The CoNLL shared tasks of 2004-2005 focused on SRL [10, 11]. In a way, those tasks set the de-facto standard dataset for training and evaluation as PropBank. The participants received preprocessed data which contained pre-segmented words, POS tags, base chunks, clauses, and named entities, in addition to full syntactic trees from two different parsers. Combined, a total of 29 systems were submitted. In this work, we present the best system of CoNLL-2005, not only because it is first, but because of its novelty and importance in future works.

In their paper, Punyakanok et al. [38] approached the task by splitting it into four sub-tasks:

- **Pruning.** Same rationale and method as Xue and Palmer.
- **Argument Identification.** Use a binary classifier to identify whether a candidate is an argument or not. The features used for the binary classifier are the ones used by Gildea and Jurafsky, among others.
- **Argument Classification.** Use a multi-class classifier to predict the types of argument candidates.
- **Inference.** A list of eight constraints over argument labeling is given, such as "arguments cannot overlap predicate" and "no duplicate argument classes", and encoded as an integer linear programming (ILP) problem, which can be solved using an ILP solver.

Punyakanok et al.'s system scored an F1 measure of 77.92.

Based on Punyakanok et al.'s constraints and ILP formulation, Täckström et al. [41] reformulated the problem as a dynamic program that finds a solution four times faster than an off-the-shelf ILP solver. The system scores comparably to state-of-the-art at the time of writing on the CoNLL-2005 dataset, and achieves state-of-the-art on the FrameNet dataset.

Zhou and Xu [48] was the first paper to introduce deep neural networks to SRL, and thus reintroducing the task as a sequence-tagging problem. Keep in mind that up until that point, end-to-end SRL systems were scarce, and did not perform as well as systems that split the problem into smaller subtasks. The neural network architecture comprised of eight bidirectional long short-term memory (BiLSTM) layers with a conditional random field (CRF) as the final layer for tag sequence prediction. Their system achieves F1 score of 81.07 on the CoNLL-2005 data (PropBank), but was not tested on FrameNet data.

Based on Zhou and Xu, He et al. [20] introduced a similar network architecture: the network is still eight layers deep, but "highway connections" and dropout were added to alleviate the vanishing gradient problem and to avoid overfitting, respectively. Instead of a CRF as the final layer, He et al. use the more common softmax function to output a tag-sequence distribution. Their system scored an F1 of 83.2 on the CoNLL-2005 data and 83.4 on the CoNLL-2012 data.

Very recently, Peters et al. [36] proposed a novel approach for word embeddings, which, when used, improves He et al.'s CoNLL-2012 F1 score, achieving 84.6.

3.2 Multilingual FrameNet

A great effort is made to expand FrameNet to other languages. As of today there are FrameNets in Finnish [24], Spanish [40], German [8], Japanese [33], Chinese [46], Korean [30], Brazilian Portuguese [43], Swedish [7], French [9], Danish [5], Polish [47], Italian [42], Slovenian [26], and of course, Hebrew [19, 37].

The methods presented in these papers are very similar: assume the universality of the English frame inventory, and under that assumption, tag, either manually or semi-automatically, sentences in the desired language using this inventory. Almost every language has its specific corner cases where the English frames are insufficient. In those cases, usually new frames are created specifically for the language in question. An example of a corner case of Hebrew is multi-word lexical units like *give up* and *turn in* - in Hebrew, these LUs might not appear as contiguous words. This case was solved by allowing annotation of discontinuous units, but add a binary flag to LUs that must appear contiguous.

3.3 Word Alignment

In this work, we aim to build a Hebrew dataset of annotated sentences by projecting annotations from English sentences. Our strategy relies on projection of lexical, syntactic and shallow-semantic frame annotations across aligned pairs of English-Hebrew sentences. One of the basis of such projection consists of identifying word alignment. Previous work in Statistical Machine Translation (SMT) has developed robust mathematical methods to identify such lexical alignment across languages. We review this work in this Section.

Given a source sentence $f_1^J = f_1, \ldots, f_j, \ldots, f_J$ and a target sentence $e_1^I = e_1, \ldots, e_i, \ldots, e_I$, a word alignment $\mathcal{A}$ is defined as

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

We begin by presenting statistical alignment models [32], followed by advancements in alignment models using neural networks.

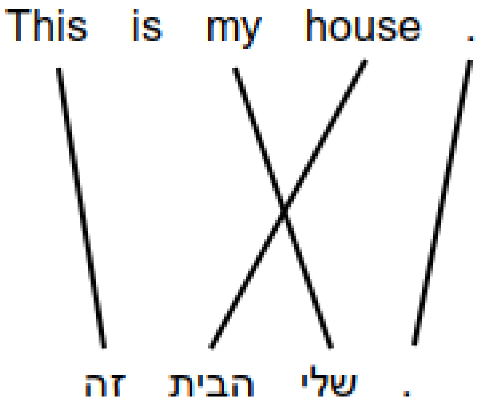
One cannot discuss statistical alignment models without mentioning machine translation. In statistical machine translation, we try to model the probability $\Pr\left(f_1^J \mid e_1^I\right)$. In statistical alignment models $\Pr\left(f_1^J, a_1^J \mid e_1^I\right)$, an alignment a_1^J representing a mapping between words in f_1^J to words in e_1^I is added. The relationship between the translation and alignment models is given by

$$\Pr\left(f_1^J \mid e_1^I\right) = \sum_{a_1^J} \Pr\left(f_1^J, a_1^J \mid e_1^I\right)$$

An alignment may map a source word to the empty word, as shown in figure 3.2.

Statistical models depend on a set of unknown parameters θ that is learned from training data. To train the unknown parameters θ, we are given a

Figure 3.2: Example of alignment to null word



parallel training corpus, *i.e.,* aligned at the sentence level, D. We find θ by maximizing the likelihood on the corpus:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \prod_{(\mathbf{f},\mathbf{e}) \in D} \sum_{\mathbf{a}} p_{\theta}\left(\mathbf{f}, \mathbf{a} \mid \mathbf{e}\right)$$

Which means that finding the best alignment comes down to the following:

$$\hat{a}_1^J = \underset{a_1^J}{\operatorname{argmax}}\, p_{\hat{\theta}}\left(f_1^J, a_1^J \mid e_1^I\right)$$

These maximizations are usually found using the expectation maximization (EM) algorithm.

Given the sets S (sure) of unambiguous alignments and P (possible) of ambiguous alignments, the quality of an alignment A is computed by appropriately redefined precision and recall measures:

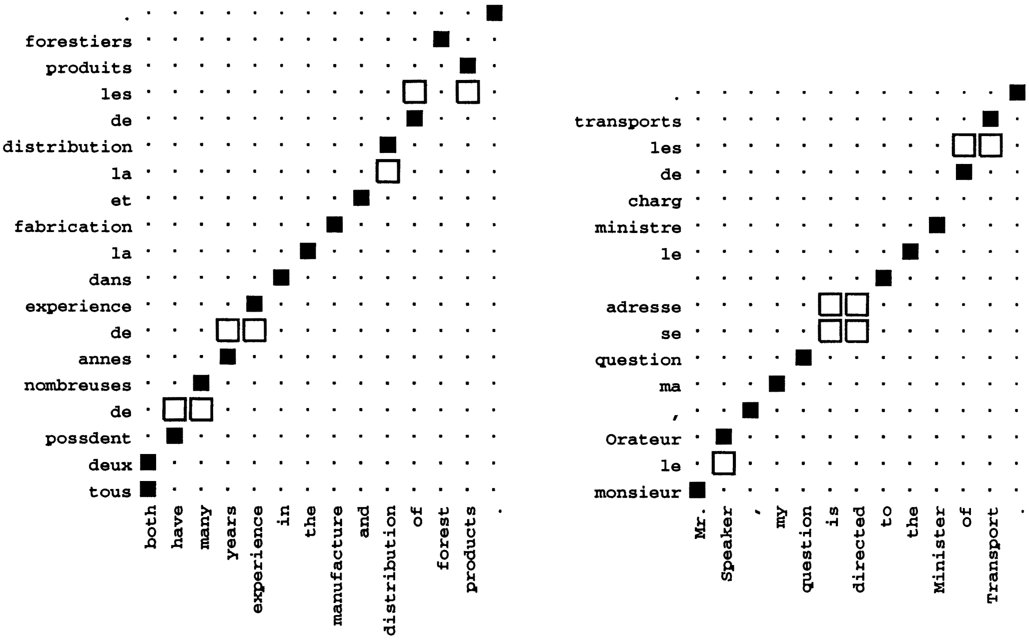
$$\text{recall} = \frac{|A \cap S|}{|S|}, \quad \text{precision} = \frac{|A \cap P|}{|A|}$$

and by the following alignment error rate (AER):

$$\text{AER}(S, P; A) = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}$$

Extrinsic (task-based) evaluations determine the quality of an alignment model by comparing a given task metric given different alignment models. For machine translation, the BLEU metric is usually the task metric [35].

Figure 3.3: Example alignment from Och and Ney [32]



In many machine learning tasks, a large dataset is key. Training an alignment model is no different, and a large corpus lowers AER significantly. Och and Ney shows this on an English-French corpus:

Corpus size	AER
0.5K	23.1
8K	18.7
128K	12.1
1470K	8.6

In addition to statistical alignment models, different methods have been introduced to learn alignment models from corpus data:

- Heuristic models which are based on similarity function.
- Statistical models based on EM.
- Fertility-based models which take into account how many target words are aligned to a source word.
- Exploiting a-priori bilingual dictionaries.

- Neural networks based on attention mechanisms which learn an alignment as a side-product of training a network for machine translation [3].

3.4 Hebrew Computational Linguistics

As if natural language processing in English is not complicated enough, NLP in Hebrew is both a blessing and a curse. A blessing because there is not much work done in Hebrew, so there are always problems waiting to be solved; A curse due to scarce resources, rich morphology, and complex word segmentation.

Itai and Wintner [22] contributed Hebrew computational linguistics corpora in Modern Hebrew, and a suite of resources and tools to process Modern Hebrew.

Adler and Elhadad [1] introduced fully unsupervised Hebrew morphological disambiguation (MD) and segmentation. Their result on a 6 million words corpus is 92.32% on part-of-speech tagging and 88.5% for MD.

Goldberg and Elhadad [17] followed with an Easy First, non-directional dependency parser that outperforms MALTPARSER, a state-of-the-art transition based parser [31], and achieves results close to those of the first-order MSTPARSER, a graph based parser [27]. Goldberg and Elhadad achieve an accuracy of 84.2% when given gold part-of-speech tags and MD, and 76.2% when part-of-speech tags and MD are given by automatic tools.

Goldberg and Elhadad [18] presented a constituency parsing system for Modern Hebrew. They show that by using a CKY lattice parsing, it is possible to do both word-segmentation and constituency-parsing jointly. When given gold segmentation, the parser achieves F1 of 85.7, and 76.95 using lattice parsing for segmentation.

In our work, we use YAP (Yet Another Parser) [29], a state-of-the-art morpheme-based tool for morphological analysis and disambiguation. It will be presented in more detail in §4.2.

4 Starting Points

Our overall strategy to develop a Hebrew FrameNet is to acquire a dataset of aligned sentence pairs English/Hebrew and then derive from them annotations on the Hebrew sentences by projecting annotations from English to Hebrew.

To this end, we identify the following prerequisites:

- Collect pairs of aligned sentences
- Compute lexical alignment across the corpus
- Perform syntactic analysis with a similar annotation scheme across English and Hebrew sentences
- Perform SRL analysis of the English side
- Project annotations from English to Hebrew

Figure 4.1 shows an example of how everything fits together.

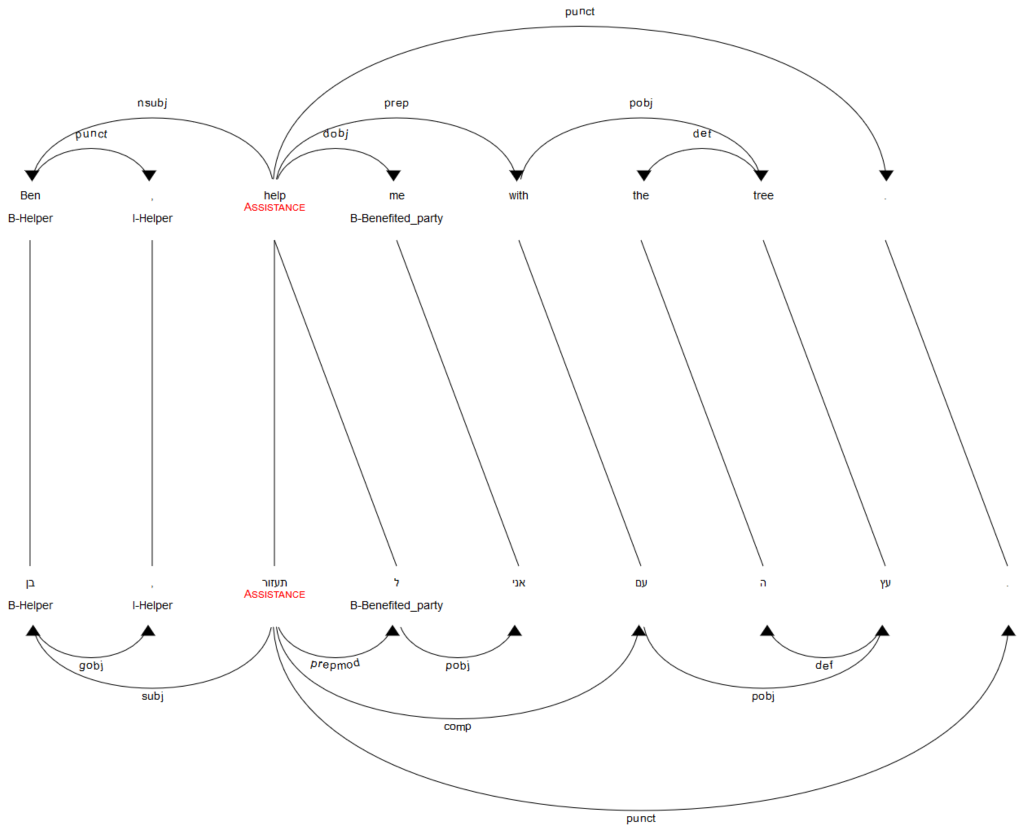
We present in the Chapter the existing tools and methods we have collected to perform these prerequisite steps.

4.1 Aligned Corpus Collection: OpenSubtitles

Lison and Tiedemann [25] collected 2.6 billion sentences in 60 different languages and 152,939 movies and TV episodes from the OpenSubtitles[1] database (mainly subtitles for pirated movies). From these monolingual languages, 1,689 bitexts were produced. Each file has meta-data such as IMDb (Internet Movie Database) identifier, when was the file uploaded to

[1] http://www.opensubtitles.org

Figure 4.1: All prerequisites visualized



OpenSubtitles.org, etc. For our needs, the dataset has 23,727,452 aligned sentence pairs English/Hebrew in varying degrees of quality.

4.2 Hebrew Syntactic Analysis with YAP

As discussed briefly in §3.4, Hebrew is not an easy language to work with. Just like with English, we want to find information such as part-of-speech tags and dependency parse trees, but unlike English, there are two steps we must perform first: segmentation and morphological disambiguation. Table 4.1 shows an example of the word בצלם and its many meanings.

On the grounds of ease of use, we decided to use YAP (Yet Another Parser) [29] for segmentation, disambiguation, and universal dependency parsing. YAP takes as input tokenized sentences, and returns a CoNLL-U formatted file. Running YAP on our corpus took approximately one month.

Table 4.1: Possible analyses for the word בצלם from Adler and Elhadad [1]

<table>
<tr><th>Segmentation</th><th>Meaning</th></tr>
<tr><td>בצלם</td><td>name of human rights association (Betselem)</td></tr>
<tr><td>בצלם</td><td>while taking a picture</td></tr>
<tr><td>בצל|ם</td><td>their onion</td></tr>
<tr><td>ב|צל|ם</td><td>under their shadow</td></tr>
<tr><td>ב|צלם</td><td>in a photographer</td></tr>
<tr><td>ב|צלם</td><td>in the photographer</td></tr>
</table>

4.3 Word Alignment with `fast_align`

Out of three word aligners, we chose `fast_align` [14] to produce word alignments from the OpenSubtitles sentence pairs. The other two aligners took significantly more time, and upon manual inspection of the output, produced inferior alignments compared to `fast_align`.

`fast_align`'s model is based on IBM Model 2 for word alignment, and is reparameterized for faster and more effective training (using a variation of EM).

Dyer et al. report a training time of 6 hours on an Arabic-English corpus of 368M tokens. On our dataset, which has 383M tokens (194M English, 189M Hebrew), training and aligning took approximately 24 hours.

181M English-Hebrew word pairs were identified, with one English word mapping, on average, to 1.25 Hebrew words.

4.4 English SRL Analysis with SEMAFOR

Out of the many automatic English SRL systems we tried, SEMAFOR [13] came out triumphant in both ease of use and overall results (based on manual inspection). It is trained on FrameNet v1.5, which is not the latest version, but seems to be what most other systems use (even ones that came after v1.7 was released).

SEMAFOR splits its task to three parts:

1. **Target Identification.** Heuristically find frame-evoking words and/or phrases ("targets") in sentences.

2. **Frame Identification.** Using a log-linear model, find a probability distribution over frames for each target, and label the target with the most probable frame.

3. **Argument Identification.** Using a second log-linear model, find the spans in each sentence that fit the selected frame, and ensure that arguments do not overlap.

It scores an F1 of 79.8 on the argument identification task alone (given gold targets and frames), and 46.49, compared to a baseline of 42.01, on the full pipeline.

4.5 Projection Strategy

Following Van der Plas et al. [44], our projection strategy is Direct Semantic Transfer (DST), where we transfer semantic relationships between a source sentence and a target sentence iff the constituents in the relationship are aligned.

This strategy is simple, and we consider a good start to tackle the problem at hand. It is important to note that Van der Plas et al. worked on DST between English and French, neither of which is a morphologically rich language, and using the high-quality Europarl dataset (proceedings of the European parliament), and PropBank-style annotations, meaning, for example, that a verb must always be aligned to another verb, which is not the case in FrameNet annotations.

5 Research Question

Our work addressed the following questions:

Given an aligned corpus of English-Hebrew sentences and English SRL annotations, can we obtain a reliable projection mechanism to Hebrew sentences?

Can we design a mechanism to automatically identify reliable pairs of sentences given the set of annotations available on the English and on the Hebrew sides? Based on this classification, we could filter the raw dataset in terms of confidence using a predictive method to measure odds of success of the projection.

We develop a manual control dataset to measure the performance of various unsupervised projection methods. How can we decide what to annotate manually and how much data is necessary to qualify the dataset produced automatically?

Our approach is empirical – we collect datasets, measure empirical distributions and report on the complexity of the task through various statistical metrics.

6 Methods

6.1 Data Collection and Filtering

The OpenSubtitles dataset provides 23.7 million English-Hebrew sentence pairs. Due to the nature of the dataset, there are pairs (sometimes entire files) that are very noisy. Examples of such noise include:

- One language has extra sentences which are not in the other language.
- OCR artefacts, *e.g.,* instead of the word "well" there is `\/\/e||`.
- Special unicode characters unrelated to the text, *e.g.,* musical note symbol to let hearing impaired viewers know there is a song playing.
- Some subtitles have a "group" associated with them, *i.e.,* the people who made the subtitles, which appears as a subtitle, causing an offset of the entire file.
- There are many joined words, either typos or OCR artefacts, such "Amanonce" instead of "A man once".

Due to the large number of sentence pairs, we have no problem getting rid of all the noisy files in the dataset. The removal of noise consists of finding files that either have no tokens left after tokenization, have more than one percent of unexpected symbols (`|`, `&`, etc.), or manually inspected and found to be noisy.

After filtering, we have 23,062,193 sentence pairs left.

6.2 Hebrew Preprocessing

The importance of Hebrew preprocessing in our work is twofold: (a) we use features from the dependency parse tree later on in the pipeline, and (b) it acts as a sanity check for our data.

The preprocessing pipeline consists of morphological analysis, morphological disambiguation, and dependency parsing. Before segmentation there were 118,236,346 tokens and 2,468,583 types, and we predict that after segmentation, the number of tokens will rise, but the number of types will be significantly lower. Indeed, after segmentation we are left with 188,375,525 tokens and 894,759 types. This is a surprising number, as one would expect the number of types in Hebrew to be approximately 200-300K, not 900K. A quick check shows that 782,247 types appear less than ten times in the entire dataset, leaving us with a more reasonable vocabulary of 112,512 types. Naturally, segmentation also affects sentence length, with 5.13 words before segmentation, and 8.17 after. Both of these numbers suggest that the sentences in the dataset are very short, most of them too short to be interesting with respect to SRL. We see a similar trend in English.

Next is a feature concerning the dependency parse tree. With an average depth of 2.96, again, the average sentence is very short and uninteresting for our goal.

The three most prominent parts-of-speech are nouns with 22,547,152 instances, personal pronouns with 13,941,874 instances, and verbs with 13,597,524 instances.

6.3 English Preprocessing

As part of SEMAFOR's pipeline, preprocessing is done on the input sentences using MaltParser [31] pre-trained on sections 02-21 of the WSJ section of the PTB.

On the English side we have 148,815,217 tokens consisting of 1,540,672 types, with sentence length averaging at 6.45. 41,304,174 of them are nouns, 36,219,769 personal pronouns, and 23,778,538 verbs. The depth of the dependency parse tree is slightly smaller than in Hebrew, averaging at 2.44.

6.4 Semantic Role Labeling

Actual output from SEMAFOR consists of 55,246,362 frames across the entire dataset, but only 784 unique frames were evoked out of the 1,020 frames available in FrameNet v1.5. On average, each sentence had 2.4 frames and 1.17 frame elements per frame. It is no surprise to find the most general frame elements being the most frequent - 3,891,862 instances of ENTITY, 3,299,307 instances of AGENT, and 3,048,768 instances of THEME.

6.5 Filtering

After preprocessing both English and Hebrew, it is time to filter the 23M sentence pairs. To this end, we built a graphical user interface that allows us to visualize the word alignment, dependency tree of both English and Hebrew, the English SRL annotations, and the Hebrew SRL projections (where successful).

124 sentences were manually annotated as one of the following six options:

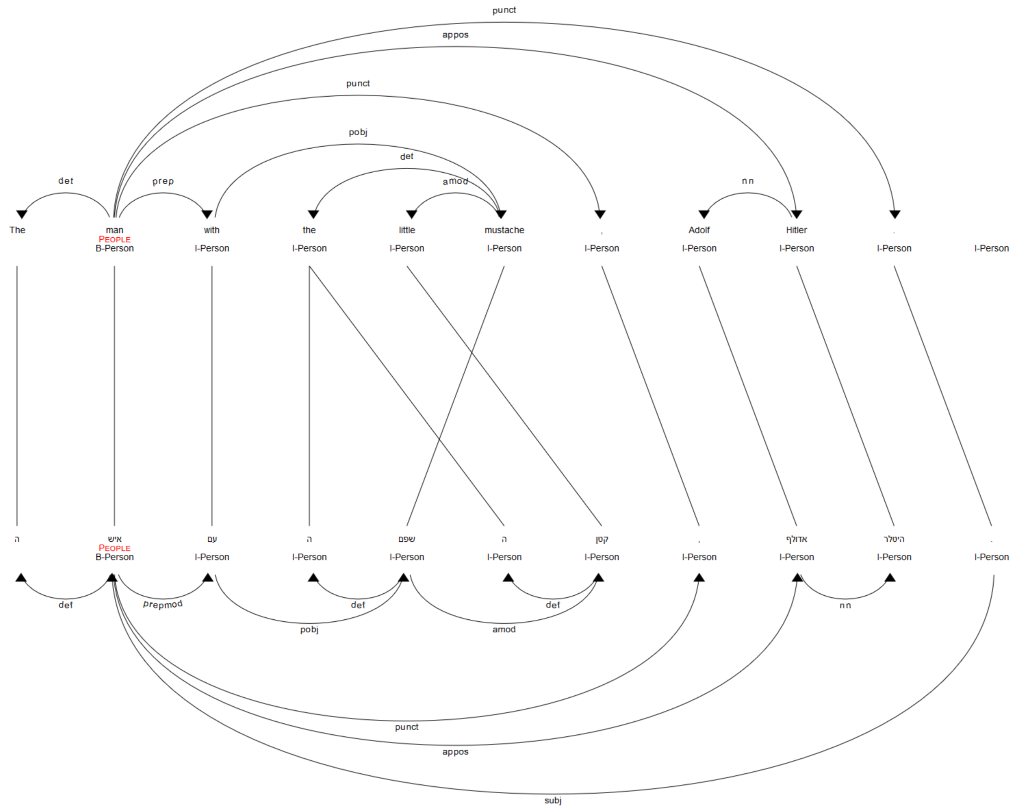
- Error in sentence alignment *
- Error in word alignment
- Poor translation *
- Poor syntactic parsing
- Poor frame parsing
- Good

The items marked with an asterisk are problems in the dataset itself, *i.e.,* better tools cannot improve these sentences' quality.

Using these annotations and random resampling, we train a linear classifier to automatically annotate other sentences. To make the problem binary classifiable, we give a label of 1 to sentences marked as "Good", and 0 otherwise. We annotate a new sentence if the classifier gives it more than 80% of being "Good". For training, we use the following features:

- IMDb rating, if available

Figure 6.1: Example of our visualization



- Sentence lengths (English/Hebrew)
- English-Hebrew sentence length ratio
- Number of frames in the English sentence
- Number of one-to-one word alignments, *i.e.*, which align one English word to one Hebrew word
- Number of one-to-many word alignments
- Dependency parse tree depth (English/Hebrew)

Surprisingly enough, we do manage to make the classifier annotate sufficiently long sentences (19.74 words per sentence, on average) with good sentence and word alignments.

6.6 Projection

The final step is a simple one - projecting the English SRL annotations on the Hebrew sentence. The projection itself finds the head word of an annotated span, takes the word aligned to the head word, find its subtree, and annotates it accordingly.

Although both theoretically and practically simple, results are very subpar, mostly due to bad word alignment. Usually, the head word of a span maps to zero or more than one word, making simple projections hard to obtain.

7 Results and Evaluation

7.1 Word Pairs

Using the word alignments that align one English word to exactly one Hebrew word, we now have an empirical dictionary with which we can automatically translate English LUs.

We have gathered about 115M word pair instances (hw, ew) with one to one mapping. These word pairs improve on existing manually curated bilingual word mappings because they cover inflected word forms as well as proper nouns which are not typically available in word lists.

The companion website of this work provides detailed statistics on the distribution of these word pairs.

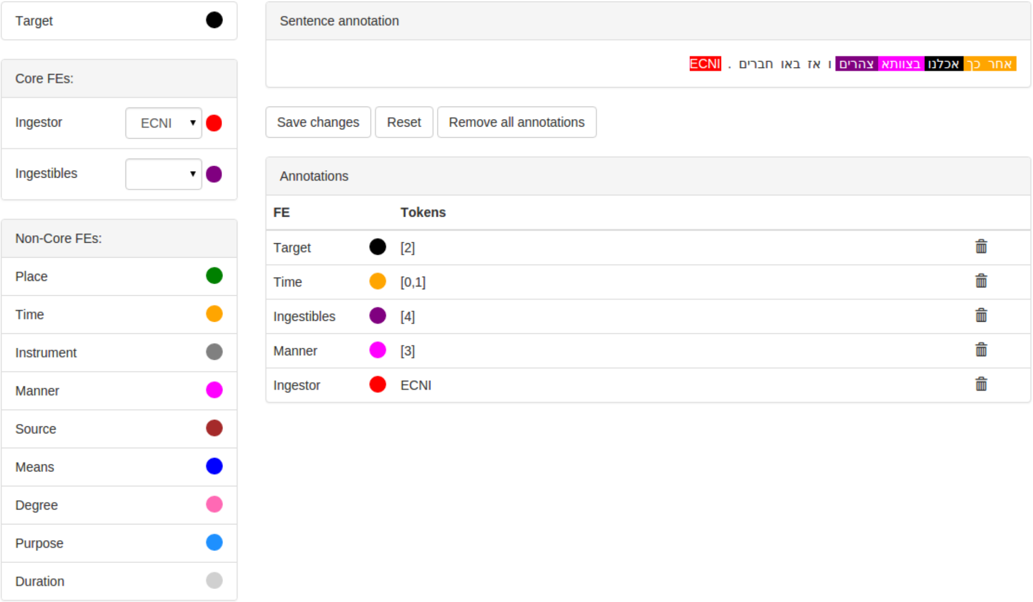
7.2 Exemplar Sentences

We manually annotated 124 sentences, 56 of which annotated as "Good". Using these annotations as a seed, the classifier (described in §6.5) fully annotated 11,205 sentences (including morphological, syntactic and SRL annotations). These sentences are candidates to serve as exemplar sentences of their corresponding frame, which we add to the Hebrew FrameNet repository, as shown in Figure 7.1.

7.3 Frames

The 11K sentences annotated by the classifier have 678 frames among them, which covers two thirds of the FrameNet v1.5 frame inventory, enabling us to further enrich the Hebrew FrameNet Project. An example of

Figure 7.1: Example of Hebrew FrameNet annotated exemplar sentence



Target ●

Core FEs:

Ingestor	ECNI ▾	●
Ingestibles	▾	●

Non-Core FEs:

Place	●
Time	●
Instrument	●
Manner	●
Source	●
Means	●
Degree	●
Purpose	●
Duration	●

Sentence annotation

אחר כך אכלנו בצוותא צהרים ו אז באו חברים . ECNI

Annotations

FE		Tokens	
Target	●	[2]	🗑
Time	●	[0,1]	🗑
Ingestibles	●	[4]	🗑
Manner	●	[3]	🗑
Ingestor	●	ECNI	🗑

such a frame is the ABANDONMENT frame, shown in Figure 7.2.

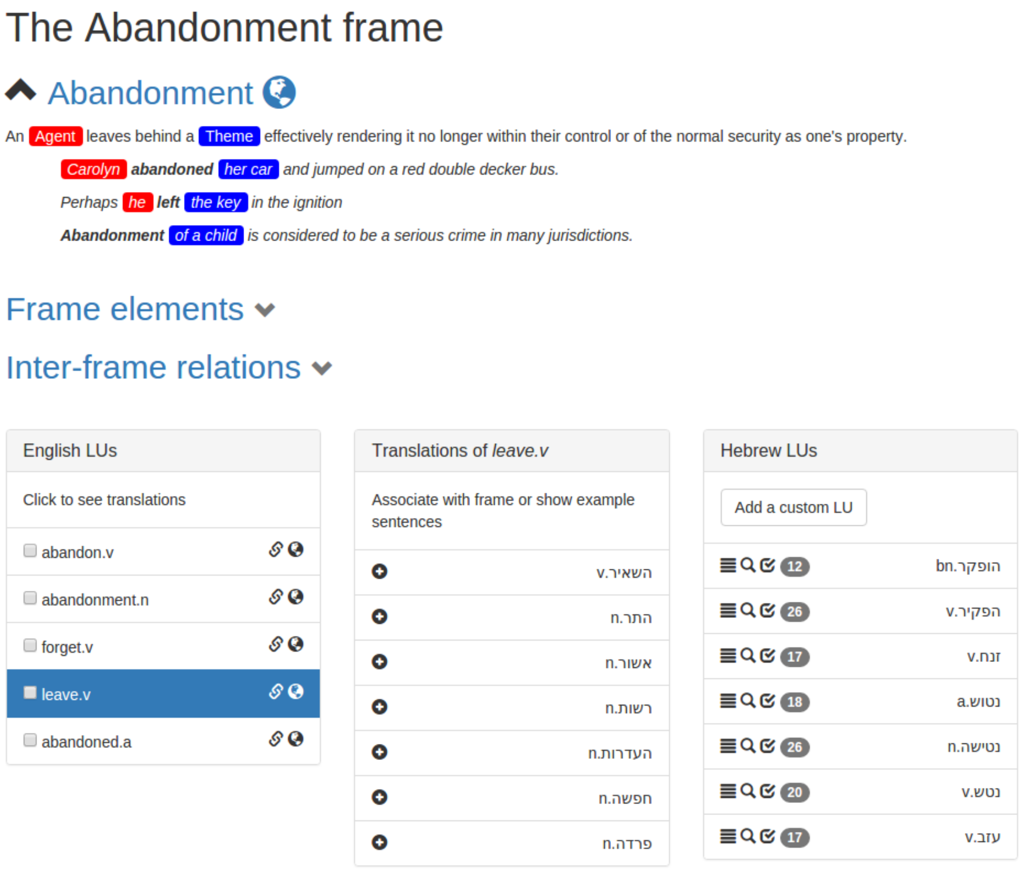
7.4 Lexical Units

Assuming one word per LU, we manage to add 5,258 LUs to Hebrew FrameNet. An example of adding a LU is shown in Figure 7.3.

A complete list of all exemplar sentences, word-pairs, frames, and LUs are available online [1].

[1] https://bgunlp.github.io

Figure 7.2: The ABANDONMENT frame



The Abandonment frame

Abandonment

An Agent leaves behind a Theme effectively rendering it no longer within their control or of the normal security as one's property.

*Carolyn **abandoned** her car and jumped on a red double decker bus.*

*Perhaps he **left** the key in the ignition*

***Abandonment** of a child is considered to be a serious crime in many jurisdictions.*

Frame elements

Inter-frame relations

English LUs

Click to see translations

- □ abandon.v
- □ abandonment.n
- □ forget.v
- □ leave.v
- □ abandoned.a

Translations of *leave.v*

Associate with frame or show example sentences

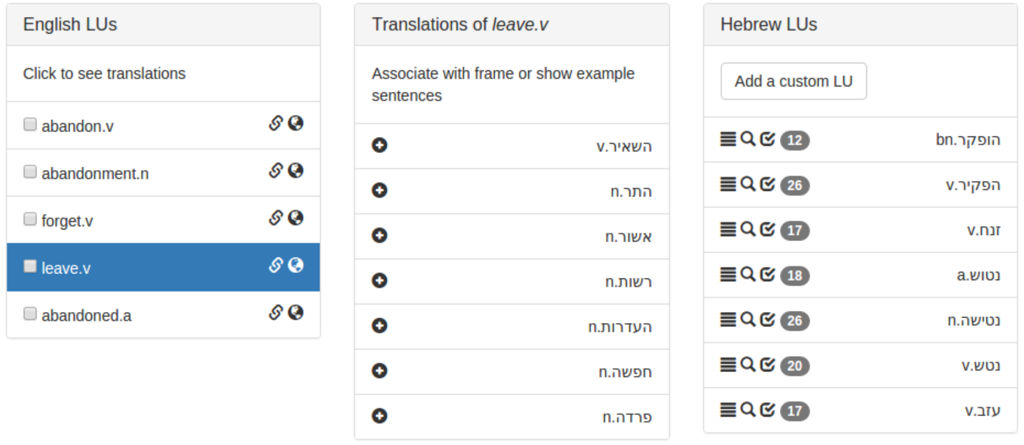
- השאיר.v
- התר.n
- אשור.n
- רשות.n
- העדרות.n
- חפשה.n
- פרדה.n

Hebrew LUs

Add a custom LU

12	הופקר.bn
26	הפקיר.v
17	זנח.v
18	נטוש.a
26	נטישה.n
20	נטש.v
17	עזב.v

Figure 7.3: Adding a LU



English LUs

Click to see translations

- □ abandon.v
- □ abandonment.n
- □ forget.v
- □ leave.v
- □ abandoned.a

Translations of *leave.v*

Associate with frame or show example sentences

- השאיר.v
- התר.n
- אשור.n
- רשות.n
- העדרות.n
- חפשה.n
- פרדה.n

Hebrew LUs

Add a custom LU

12	הופקר.bn
26	הפקיר.v
17	זנח.v
18	נטוש.a
26	נטישה.n
20	נטש.v
17	עזב.v

8 Conclusion

The conclusion of this work leaves us ambivalent. We have shown that working with both a noisy dataset and the Hebrew language is, indeed, no easy task, and the results leave much to be desired. But, there is light at the end of the tunnel, as improvements of different parts of the pipeline could be beneficial to its entirety.

Even in the presence of much noise at all stages of our analysis pipeline, the mass of data available in the OpenSubtitles project allowed us to extract promising data to fill the Hebrew FrameNet project with 678 frames (out of about 1,000 frames in the FrameNet inventory), annotated with over 11K fully annotated Hebrew sentences (morphological, syntactic and semantic role annotations) and with over 5K lexical units associated.

These figures provide sufficient quantity to enable training an SRL system in Hebrew.

We still need to assess the quality of the automatically produced data that we have pushed into Hebrew FrameNet. The Hebrew FrameNet tool provides data curation methods that will help in this objective. Our experiment with annotating a seed of only about 100 sentences to determine sentence quality is encouraging in indicating practical methodology to identify different aspects of the data produced (LUs and Frame Annotations).

In the future, we also seek to try and improve each of the steps in the pipeline we have deployed, and implement a variation of Akbik et al. [2] that is relevant to FrameNet, along with trying more recent automatic frame-semantic parsers.

Bibliography

[1] Adler, M. and Elhadad, M. (2006). An unsupervised morpheme-based hmm for Hebrew morphological disambiguation. In Calzolari, N., Cardie, C., and Isabelle, P., editors, *ACL*. The Association for Computer Linguistics.

[2] Akbik, A., Danilevsky, M., Li, Y., Vaithyanathan, S., Zhu, H., et al. (2015). Generating high quality proposition banks for multilingual semantic role labeling. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 397–407.

[3] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[4] Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

[5] Bick, E. (2011). A framenet for danish.

[6] Bonial, C., Hwang, J., Bonn, J., Conger, K., Babko-Malaya, O., and Palmer, M. (2012). English propbank annotation guidelines. *Center for Computational Language and Education Research Institute of Cognitive Science University of Colorado at Boulder*.

[7] Borin, L., Dannélls, D., Forsberg, M., Gronostaj, M. T., and Kokkinakis, D. (2010). The past meets the present in swedish framenet++. In *14th EURALEX international congress*, pages 269–281.

[8] Burchardt, A., Erk, K., Frank, A., Kowalski, A., Padó, S., and Pinkal, M. (2006). The salsa corpus: a german corpus resource for lexical semantics. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-2006)*, pages 969–974.

[9] Candito, M., Amsili, P., Barque, L., Benamara, F., De Chalendar, G., Djemaa, M., Haas, P., Huyghe, R., Mathieu, Y. Y., Muller, P., et al. (2014). Developing a french framenet: Methodology and first results. In *LREC-The 9th edition of the Language Resources and Evaluation Conference*.

[10] Carreras, X. and Màrquez, L. (2004). Introduction to the conll-2004 shared task: Semantic role labeling.

[11] Carreras, X. and Màrquez, L. (2005). Introduction to the conll-2005 shared task: Semantic role labeling. In *Proceedings of the ninth conference on computational natural language learning*, pages 152–164. Association for Computational Linguistics.

[12] Christensen, J., Soderland, S., Etzioni, O., et al. (2010). Semantic role labeling for open information extraction. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pages 52–60. Association for Computational Linguistics.

[13] Das, D., Chen, D., Martins, A. F., Schneider, N., and Smith, N. A. (2014). Frame-semantic parsing. *Computational linguistics*, 40(1):9–56.

[14] Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. Association for Computational Linguistics.

[15] Fillmore, C. J. (1976). Frame semantics and the nature of language. *Annals of the New York Academy of Sciences*, 280(1):20–32.

[16] Gildea, D. and Jurafsky, D. (2002). Automatic labeling of semantic roles. *Computational linguistics*, 28(3):245–288.

[17] Goldberg, Y. and Elhadad, M. (2010). Easy first dependency parsing of modern hebrew. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 103–107. Association for Computational Linguistics.

[18] Goldberg, Y. and Elhadad, M. (2013). Word segmentation, unknown-word resolution, and morphological agreement in a hebrew parsing system. *Computational Linguistics*, 39(1):121–160.

[19] Hayoun, A. and Elhadad, M. (2016). The hebrew framenet project. In *LREC*.

[20] He, L., Lee, K., Lewis, M., and Zettlemoyer, L. (2017). Deep semantic role labeling: What works and what's next. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 473–483.

[21] He, L., Lewis, M., and Zettlemoyer, L. (2015). Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 643–653.

[22] Itai, A. and Wintner, S. (2008). Language resources for hebrew. *Language Resources and Evaluation*, 42(1):75–98.

[23] Kingsbury, P. and Palmer, M. (2002). From treebank to propbank. In *LREC*, pages 1989–1993. Citeseer.

[24] Lindén, K., Haltia, H., Luukkonen, J., Laine, A. O., Roivainen, H., and Väisänen, N. (2017). Finnfn 1.0: The finnish frame semantic database. *Nordic Journal of Linguistics*, 40(3):287–311.

[25] Lison, P. and Tiedemann, J. (2016). OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

[26] Lönneker-Rodman, B., Baker, C., and Hong, J. (2008). The new framenet desktop: A usage scenario for slovenian. *Programme Committee 7*, page 147.

[27] McDonald, R., Crammer, K., and Pereira, F. (2005). Online large-margin training of dependency parsers. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 91–98. Association for Computational Linguistics.

[28] Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., and Grishman, R. (2004). The nombank project: An interim report. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*.

[29] More, A. and Tsarfaty, R. (2016). Data-driven morphological analysis and disambiguation for morphologically rich languages and universal dependencies. In *Proceedings of COLING 2016*.

[30] Nam, S., Park, J., Kim, Y., Hahm, Y., Hwang, D., and Choi, K.-S. (2014). Korean framenet for semantic analysis. In *Proceedings of the 13th International Semantic Web Conference*.

[31] Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

[32] Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Comput. Linguist.*, 29(1):19–51.

[33] Ohara, K. H., Fujii, S., Ohori, T., Suzuki, R., Saito, H., and Ishizaki, S. (2004). The japanese framenet project: An introduction. In *Proceedings of LREC-04 Satellite Workshop "Building Lexical Resources from Semantically Annotated Corpora"(LREC 2004)*, pages 9–11. Citeseer.

[34] Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

[35] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

[36] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. *ArXiv e-prints*.

[37] Petruck, M. R. (2005). Towards hebrew framenet.

[38] Punyakanok, V., Roth, D., and Yih, W.-t. (2008). The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2):257–287.

[39] Ruppenhofer, J., Ellsworth, M., Petruck, M. R., Johnson, C. R., and Scheffczyk, J. (2016). *FrameNet II: Extended theory and practice*. Institut für Deutsche Sprache, Bibliothek.

[40] Subirats, C. and Petruck, M. (2003). Surprise: Spanish framenet. In *Proceedings of CIL*, volume 17, page 188.

[41] Täckström, O., Ganchev, K., and Das, D. (2015). Efficient inference and structured learning for semantic role labeling. *Transactions of the Association for Computational Linguistics*, 3:29–41.

[42] Tonelli, S. and Pianta, E. (2008). Frame information transfer from english to italian. In *LREC*.

[43] Torrent, T. T. and Ellsworth, M. (2013). 3) behind the labels: Criteria for defining analytical categories in framenet brasil. *Revista Veredas*, 17(1-).

[44] Van der Plas, L., Merlo, P., and Henderson, J. (2011). Scaling up automatic cross-lingual semantic role annotation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 299–304. Association for Computational Linguistics.

[45] Xue, N. and Palmer, M. (2004). Calibrating features for semantic role labeling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

[46] You, L. and Liu, K. (2005). Building chinese framenet database. In *Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference on*, pages 301–306. IEEE.

[47] Zawisławska, M., Derwojedowa, M., and Linde-Usiekniewicz, J. (2008). A framenet for polish. In *Converging Evidence: Proceedings to the Third International Conference of the German Cognitive Linguistics Association (GCLA'08)*, pages 116–117.

[48] Zhou, J. and Xu, W. (2015). End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1127–1137.