One Who Has Acquired a Good Name Has Acquired Something for Himself[1]: Named Entity Recognition on Talmudic Texts

Presented to the S. Daniel Abraham Honors Program in Partial Fulfillment of the Requirements for Completion of the Program

Stern College for Women

Yeshiva University

December 27, 2021

Adina Bruce

Mentor: Professor Joshua Waxman, Computer Science

[1] Pirkei Avot 2:7

Table of Contents

Figures

Tables

Equations

Abstract

In this paper, I will explore the intersection between Natural Language Processing and Talmudic texts. I worked with Professor Joshua Waxman at the Stern Natural Language Processing Lab during this research project to create a Named Entity Recognizer that could be used on Talmudic texts. This process included the creation of gazetteers, that is, lists of people and place names that are found in the Talmud and the Bible. The gazetteers were created through data extraction from the Jastrow Dictionary and the Brown-Driver-Briggs Dictionary using Sefaria's MongoDB database and utilizing the Compass Client and regular expressions. The gazetteers were used in the tagging of Talmudic texts which were then passed into a Naive-Bayes model Named Entity Recognizer as training data. Features such as the words surrounding each Named Entity, suffixes and prefixes, as well as a gazetteer lookup, were generated for the training data used on the model.

As part of this research, I will present a survey of the current state of the art research of using Natural Language Processing for Hebrew language texts, and especially on rabbinic texts. The Hebrew language has certain features that present challenges to utilizing popular Natural Language Processing techniques and tools that have already been developed for languages such as English. Furthermore, Hebrew from different time periods and historical sources for texts will have slight differences in grammar, sentence structure and vocabulary. Therefore, work done creating Natural Language Processing tools for Hebrew from one time period will need to be adapted in order to be used on a text from a different time period. However, techniques developed to address certain aspects of the Hebrew language, such as its high morphological ambiguity, developed for texts from any time period, are helpful to examine, to see what common challenges researchers face and what solutions are developed in the Natural Language Processing field.

Introduction

Natural Language Processing (NLP) is a field of Artificial Intelligence (AI) that deals with the issue of how computers should interpret human languages. NLP also deals with the problem of how information can be extracted from examples of human languages, such as from texts. The issue of interpreting human languages is particularly difficult for a computer as common features of human languages include ambiguity, double meanings and euphemisms. A typical NLP pipeline might look like Figure 1.

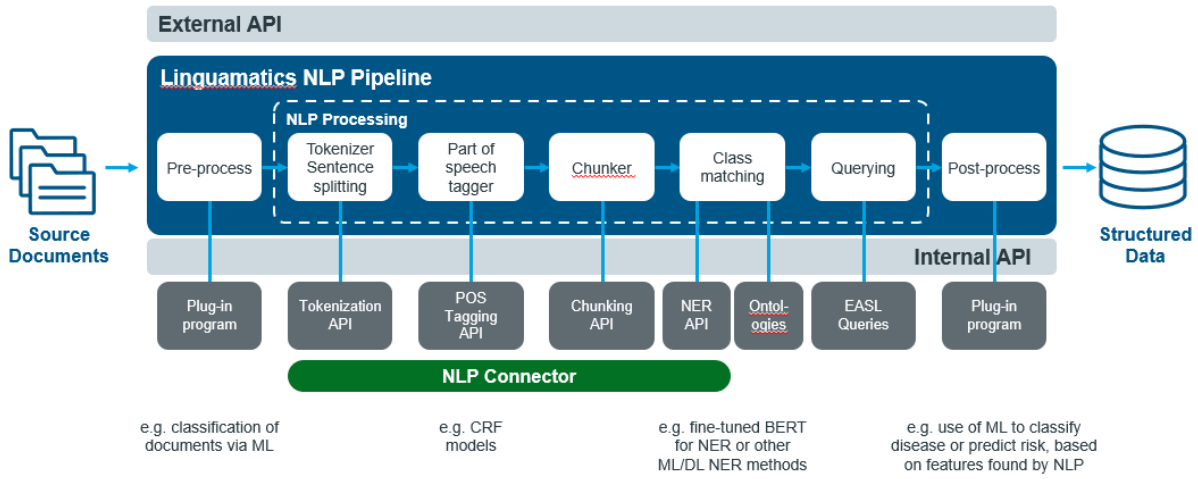

Figure 1: A NLP Pipeline[2]

There are many parts of a NLP pipeline, which will process a Natural Language into a structure where information extraction is enabled for a computer. Sentence Segmentation is the process of breaking down a large text into smaller sections that are easier to process. Word Tokenization is the further breaking down of a sentence into individual words or phrases. Parts of speech (POS) prediction is a process that associates each token with a linguistic category such as noun, pronoun, verb, adverb, adjective, preposition, conjunction and interjection. Morphological analysis is the breaking down of a word into its fundamental meaning, sometimes breaking up a word into prefixes, suffixes, and stems or root meanings. Constituency Parsing is the creation of a parse tree based on the syntactical structure of a

[2] https://www.linguamatics.com/how-does-nlp-work

sentence. Dependency Parsing also creates a parse tree, but this one is based on the dependencies of the words on each other. Named Entity Recognition is the tagging of words that have a proper name and classifying them into what kind of category they are, such as people, places or organizations. Relation Extraction describes the interactions between different Named Entities. The processing of a text might use some but not all the techniques described above.

Natural Language Processing techniques can be applied to any human language. Natural languages are defined as symbolic representations which have been created naturally by humans (Kareem Darwish, 2021). Examples of Natural Languages include English, American Sign Language, and Mandarin Chinese. While much progress has been made in the field of computational linguistics to create tools for languages such as English or Chinese, other languages such as Hebrew presents its own challenges which makes applying popular NLP tools difficult. Hebrew has high morphological ambiguity: defined as having words that often have multiple meanings associated with the same form of the word. This ambiguity makes parsing the words into different parts of speech and extracting information more complicated compared to other languages. Another issue is that there is a disparity between resources developed for some languages over others. High-resource languages, such as English and Japanese have had a lot of research and tools developed. A low- resource language such as Aramaic and Hebrew has less tools developed for its use, and often popular tools have only been developed with high-resource languages in mind.

My research focused on developing a machine learning model that would be able to apply Named Entity Recognition (NER) on a Talmudic text. The Talmud Bavli is written in Biblical and Mishnaic Hebrew, Eastern Aramaic (Neusner, 1990) and contains words from other languages such as Greek and Persian. The purpose of the NER was to tag people and places named within Talmudic texts. This was achieved through developing gazetteers that

were created by gathering data from the Jastrow Dictionary (Jastrow, 1926) and the Brown-Driver-Briggs (BDB) Lexicon (Brown, 1996) from Sefaria's MongoDB, as well as a list of Rabbinic names developed by Michael Satlow and Michael Sperling. These gazetteers were used to programmatically tag the text. The tagged text, along with additional features generated for each individual word, was used to train a Named Entity Recognizer using a Naïve-Bayes machine learning model.

Many parts of this research overlapped with other work done in the Digital Humanities and Hebrew NLP fields, especially relating to projects of data processing and machine learning which aim to analyse modern and ancient Hebrew texts. The gathering of location names into the form of a gazetteer has much use, especially relating to Authority Files used by Libraries and their databases. The listing of people found in the Talmud has very old roots going back to the *Sherira Gaon* in the 10th century, though more comprehensive and advanced work has developed over time and has gained traction in popularity in modern times. Although there have been several tools developed to process Hebrew texts using machine learning techniques, some of these tools are trained using only Modern Hebrew which limits the extent of their use on Talmudic texts.

Background

Machine Learning on Hebrew Texts

One of the main issues that comes with attempting to do NLP processing on Hebrew languages is that Hebrew has high morphological ambiguity. Other languages have less ambiguity so common NLP tools are potentially suboptimal to be used on Hebrew texts. One example of a popular NER module is spaCy, an open-source software library that specializes in NLP tools. SpaCy does have a Hebrew package that enables word tokenization but does not have one with Hebrew NER already implemented. However, spaCy does allow for

training a NER model yourself.

Because of the lack of readily available software from popular NER packages, attempts have been made to create tools specific for the Hebrew language. ONLP as described in "What's Wrong with Hebrew NLP? And How to Make it Right" (Klein, 2019) uses YAP - Yet Another (Natural Language) Parser - to accurately parse and tag Hebrew. A challenge in using general parsers and taggers on Hebrew texts is that Hebrew words are often made up of many small tokens that must be broken up in order to tag the POS. The issue of doing this, however, is that breaking up the word increases the morphological ambiguity. Additionally, the meaning of the tokens can only be understood alongside the context of the word. General parsers generally use linear pipelines to pre-process. This means that once something has been tagged in a certain way it will not be modified correctly later in the tagging process. Using an alternative strategy, ONLP uses joint morpho-syntactic parsing where the training utilizes splitting up of the different parts of speech, as well as choosing the correct morphology at the same time.

The creators of YAP claim that the revolutionary techniques used to improve Hebrew tagging will substantially help progress academic work in the Hebrew NLP field forward. However, Waxman believes from his experience of using the program on rabbinic texts that the output is not accurate. Talmudic texts, as well as other rabbinic texts, are primarily made up of Hebrew; however, the grammar, as well as words, used are different from Modern Hebrew. As the YAP tool is trained using a process that specifically works for Semitic languages it is likely that YAP would work better once trained on a wider array of Hebrew texts, dating back to earlier than Modern Hebrew.

Dicta is a non-profit research organization that specializes in creating NLP and ML tools to be used with Hebrew texts. Their web API makes available different tools developed for Hebrew texts. One tool, called Nakdan, can diacritize or add *nekudot* (vowels) to non-

voweled Hebrew words. What is particularly noteworthy about this API is its capability to allow for the input of texts such as Modern, Rabbinic or Poetic Hebrew. Unlike YAP, the Nakdan tool was created specifically in mind to be able to handle Rabbinic texts and not just Modern Hebrew texts.

Nakdan was trained using a mixture of both POS tagging, tables that generate all possible diacritizations, and then produce the ranking of the most likely diacritizations given the intersection of the previous tagging and tables created (Shmidman A. S., 2020). The training for the POS-tagging and subsequent morphological disambiguation is enacted through a 2-layer bi-LSTM (Bi-directional long short term memory) transducer trained on manually annotated corpora. Unlike YAP, Dicta's model does not perform character level tagging, but instead tags on a less granular level. Shmidman claims that this "force[s] the system to make a more logical morphological determination" and will not overfit the prediction based on the training corpus.

The Nakdan tool is distinct due to the training done on Rabbinic texts along with the specific feature that enables one to diacritize Rabbinic Hebrew. This was achieved by training the model on a specialized "historical Hebrew corpus," using a collection of "Jewish legal writings and commentaries from the 3rd-12th centuries." The historical corpus is made up of about half the fine-grained morphological tokens than the Modern Hebrew corpus, however it is composed of more words with diacritization compared to the Modern Hebrew corpus. Included in the diacritized wordlist are Babylonian Aramaic words. According to tests done by Shmidman et al., the word accuracy of Nakdan's diacritization compared to others in the field, Morfix and Snopi, is much higher. However, the test done to prove this is based on the Beit Yosef, written by Rav Joseph Karo in the late 15th century, a relatively later work compared to other Rabbinic texts. More tests done with earlier works, or primarily Aramaic works, such as the Palestinian or Babylonian Talmud, might provide less accurate results.

Further research done by the Dicta team looks to address the issue of homographs (Shmidman A. a., 2020). Models that work on Hebrew texts struggle with homographs because Hebrew is a Morphologically Rich Language (MRL). This means that the disambiguation that comes from homographs is additionally challenging for a POS tagger. One example brought by Shmidman is the word form הרי. This word can have on case where it is הֲרֵי (indeed) which is a conjunction or interjection, and another case הָרֵי (mountains) which is a noun in the masculine plural form. Training a model that disambiguates homographs can also be challenging since there is not an even distribution of the different homograph cases in the corpus data; some examples are found much more frequently than others. Training initiated to correctly tag the homographs uses corpora that attempt to target homographs that contain an imbalance of examples and include more sentences with the unbalanced homographs.

YAP was initially used on 21 cases of common homographs, some of which are extremely unbalanced in terms of use in the general corpora. The results showed that particularly for the cases of unbalanced homographs, YAP performed poorly. Implementing their own method, the Dicta team used a 2- layer Multi Layer Perceptron (MLP) utilizing context as an input. The specialized classifiers created by the Dicta team had superior performance on the homograph test cases and were more adept when used on the unbalanced words. Shmidman concludes that for specific cases where there are extremely unbalanced homographs within the corpora "specialized contrast sets are needed … in order to train effective classifiers."

This problem is particularly relevant for a NER on Talmudic texts. On top of Hebrew and Aramaic being highly ambiguous languages, many named entities overlap with other common words found in the Talmud. For example, there is a person in the Talmud known as *Rav* and another known as *Rebbi*, yet these two names are also often the beginning of the

titles of other people in the Talmud. Another pertinent example is Elisha Ben Avuya who is often cited using the term "*Acher*"- (Other) due to his later rejection of Judaism. However, "*acher*" is an extremely common word used in the Talmud, such as in the phrase "*dvar acher,"* used to bring an alternative explanation to the one previously cited. Therefore, implementing a NER for Talmudic texts that can properly handle common homographs which have multiple definitions and syntaxes is essential.

The issue of Hebrew NER is grappled with in the article "A graph database of scholastic relationships in the Babylonian Talmud" (Waxman, 2021), using statement alignment of Hebrew and English texts. NER was first executed on English text, using capitalization as well as a database of English names. The tagged Named Entities were then projected back onto the Hebrew text. Sefaria's Talmudic text uses the The Noé Edition Koren Talmud Bavli. The English translation is a direct translation of Rabbi Shteinzalt's translation of the Aramaic Hebrew into Modern Hebrew. Additionally, Shteinzalt's interpretation of the text is embedded in his translation. The exact translation is retained using bolded words, while the explanations and subtext, added to aid in the reader's understanding, are not bolded. The punctuation in both the Hebrew and English text is the same, allowing both texts to be aligned, and the Named Entity Tagging, more easily done on English texts, is then able to be projected back onto the Hebrew.

Gazetteers, Prosopography and Authority Files

Traditionally, a Gazetteer is defined as a geographical index or dictionary, typically containing information pertaining to the place, such as location, altitude or population size[3]. In machine learning, the term gazetteer is used synonymously with "lexicon", "dictionary" or "list" (Nadeau, 2007). In NER a gazetteer can be used as a lookup feature. If the word being

[3] https://en.wikipedia.org/wiki/Gazetteer

classified, as either being a named entity or not, is included in a list of named entities, it is more likely that the word is in fact a named entity. However, inclusion or non-inclusion does not mean that the word is certainly a Named Entity; there might be a named entity which was emitted from the gazetteer, or a word that is a Named Entity in some situations but not in this context. Nonetheless, using a gazetteer to look up the word can be used as one of the features used in a machine learning model.

Modern day gazetteers have been created in order to leverage digital databases of place names for use in other fields of the Digital Humanities. Through analysing Hebrew texts, the historical gazetteer Kima (Rusinek, 2021) tracks the names of geographical locations over time. The information for this gazetteer is gathered from the NLI catalogue, as well other Hebrew databases. Using similar databases, geographical gazetteers can be created to then be used for look-up feature generating. Examples of texts which collect information on geographical locations in the Talmud include *Carta's Atlas of the Period of the Second Temple, the Mishnah and the Talmud* (Avi-Yonah, 1966), *Beiträge zur Geographie und Ethnographie Babyloniens im Talmud und Midrasch* (Berliner, 1884), *La géographie du Talmud* (Neubauer, 1868) and the *Entsiḳlopedyah le-ge'ografyah Talmudit : be-tseruf mapot tsilumim ye-luḥot* (Ne'eman, 1970).

While gazetteers are typically associated with geographical locations, my research uses the format of a lookup list with people and locations. Lists of personalities from the Talmudic texts have a rich history within Jewish historical texts. The topic of prosopography, the historical and social connections between people, is particularly important within the Talmud as it proves the continuation of tradition within the *Halachic* process.

One of the earliest examples of such compilations is The *Iggeres* of *Rav Sherira Gaon* (Sherira ben Hanina, 1988). This letter was written by *Rav Sherira Gaon*, the head of the Yeshiva of *Pumbedisa* in Babylon, to the community of Kairouan, Tunisia, to answer their

questions about the history and authority of the Oral Torah. *Sherira Gaon* goes through the methodological development of the Oral Law, but also includes a chronological list of the Rabbis that appear in the Talmud.

A later work which further developed the field of genealogy and biography regarding the Rabbis of the Talmud is *Seder Hadorot* (Heilprin, 1961). The complete work aims to address a historical investigation into chronologically tracing people of Jewish importance. The book is divided into three sections: 1. The order of generations from ancient times to the 17th century, 2. The rabbis of the Talmud in alphabetical order, and 3. Books and their authors written after the Talmudic period (Zinberg, 1975). Included in the entries in part two, *Seder Tannaim V'Amoraim*, is further information associated with each Rabbi found in the Talmud, such as the sayings attributed to them and their familial relations. This work is especially noteworthy, as at the time that it was written historical work was unpopular, while *pilpul,* extremely fastidious legal discussions, was favored instead.

As the enlightenment gained traction within Jewish thought, along with a movement known as The *Haskalah,* these kinds of historical and scientific projects became more respected and popular. One such person who pursued this topic of research is Rabbi Aaron Hyman who lived in London in the early 20th century. His work *Beit Va'ad la-Ḥakhamim* (Hyman, Sefer Bet Ṿa‘ad La-Ḥakhamim, 1902) later expanded to *Oẓar Divrei Ḥakhamim u-Fitgameihem* (Hyman, Otsar Divre Ḥakhamim U-Pitgamehem, 1947) collects the sayings of Rabbis in the Talmud by topics and provides citations for their source. Additionally, his work *Toldot Tennaim V'Amoraim* (Hyman, Sefer Toldot Tana‘im Ṿe-Amora‘im, 1964) is like Heilprin's, providing a biographical dictionary of Rabbis from the Talmud, along with citations for places where they are mentioned and how they relate to other people in the Talmud.

As time went on academic methods became more acceptable for Jews to utilize when writing works about the Torah and the Talmud. Mordechai Margalioth is a person who exemplifies this shift in strategy that people utilized to study Jewish works. He studied and taught Rabbinic Literature at Hebrew University, and taught at the Jewish Theological Seminary (Margaliot (Margulies), Mordecai , 2021). His work *Enẓiklopedyah le-Ḥakhmei ha-Talmud ve-ha-Ge'onim* (Margalioth, 1945) is a scholarly publication of a biographical dictionary of Rabbinic figures found within the Talmud.

More recent work on the topic of gathering Talmudic figures in academic settings comes in the form of databases. The work done with these databases may also focus on gathering additional data related to the rabbis, other than just the name. For example, Jacob Parker's "Sages of the Talmud" (Parker, 2005) is a searchable database of more than 250 Talmudic Sages. It also holds data on the generation of each rabbi, and the student/ teacher relationship between them (Waxman, 2021).

Another example of a database developed on Rabbinic figures in the Talmud is the *Otzar Hadmuyot* (Bonayich , 2021), Index of Rabbinic Sages, by Bonayich. Bonayich is an educational institute that develops resources to aid in studying Mishnah and Gemara. The database developed includes "all spelling variations of their names, identification of their generation and place of activity, their family and scholarly relationships with other personalities, and all scholarly research written about each figure." This database contains almost 3000 entries, with much of the information on each entry constructed from the works of Hyman and Margoliath (Zhitomirsky-Geffet, 2018).

Another rabbinic database developed by Satlow and Sperling (Sperling) is partially based on the Bonayich database. Other sources from which their database is based is Hyman's book and a list compiled by Sperling. Their database includes a unique ID for each distinct person included. This is needed as also included in their database are multiple ways

of spelling the name of many of the Rabbis. Additionally, some personalities included might have multiple names, therefore the ID links the different entries to one unique personality.

Many of these databases and projects which work to gather personalities from the Talmud focus on Rabbinic personalities. This leaves gaps in the databases of all personalities mentioned in the Talmud, including ones which might have provided *halachic* input. For example, Yalta, the wife of Rav Nachman and daughter of the *Reish Galuta* and an infamous character in the Talmud for her fiery personality, is not found in Margoliath. Furthermore, other biblical personalities who are sometimes characters in the Talmud are not included in these texts. For example, the biblical figure Moses is also not found in Margoliath, despite also having interactions with Rabbis in the Talmud (*Menachot* 29b).

Despite the gaps in Talmudic personalities in these databases, it is still useful to have information on the Rabbinic figures. Sefaria uses information from the work of Satlow and Sperling, to hyperlink the names of rabbinic figures to biographies of those rabbis. Furthermore, most people mentioned in Talmudic texts will be citations of legal opinions of Rabbis, as well as the quotations from other Rabbis that support the teachings. Therefore, it is likely that any work done on Talmudic personalities that uses a database based on Rabbinic persons will cover most of those instances.

Another example of a database that gathers Talmudic and Biblical Named Entities is the *Elyonim veTachtonim* project which recently published two databases on supernatural entities in the Bible and Talmud (Kosior, 2021). Although not "people" per say, many entries included in the two databases are characters, or named beings, which are referred to in the Biblical and Talmudic texts. An example of an entry in the Talmudic database is Satan, defined in the database as an "angel". Each entry is also accompanied with a place in the text where the entity is included, in this case as "Forthwith, **Satan** stood up against Israel;25 and it is further written, He stirred up David against them saying, Go, number Israel", as quoted

in *Berakhot* 62b, with the entity bolded. Although many entries are useful as being Named Entities, others are arguably so. For example, the term *batqol* is recorded as an entity from Berakhot 3a, however, arguable a *batqol* is not an entity in of itself, but a description of a kind of voice that represents God.

Methodology

The first step of being able to label Named Entities, either programmatically or using a machine learning model, is to create a gazetteer of places and people. The gazetteer is labeled either with "LOC" for Location or "PER" for Person. The data used to develop the gazetteer was gathered from Sefaria's open-source database. As well as being an online open-source library of Jewish texts, Sefaria also makes its database available to be downloaded from github (Sefaria github).

Before doing search queries on the Sefaria database, I first explored the entries found in the Jastrow and Brown-Driver-Briggs dictionaries as they appear on the Sefaria website and on the Compass client of Sefaria's MongoDB database.

Figure 2 shows an example of one such entry:

מָתָא מ' מְחַסְיָא pr. n. pl. *Matha M'ḥasia* (or *Maḥseia*, v. **Jer. XXXII, 12**), prob. a suburb of Sura (v. Berl. Beitr. z. Geogr., p. 45, sq.). **Keth. 4ᵃ. Ber. 17ᵇ** בני מ' מ' וכ' the (gentile) inhabitants of M. M. are obstinate. **Kidd. 33ᵃ. B. Kam. 119ᵇ. Snh. 7ᵇ; Hor. 3ᵇ**

Figure 2: Jastrow Entry

Mata Mehasya is a town mentioned in the Talmud, located in southern Babylon, near the city of *Sura* (Solomon Schechter, n.d.). The academy of *Rav Ashi,* a well-respected Torah scholar, was located there. The entry is labeled as "מחסיא" as well as "'מ מתא" indicating that the place is found in Talmudic texts as both *Mehasya* and *Mata Mehasya*. The entry is

labeled as pr. n. pl. which corresponds to “proper noun of a place”[4]. A definition derived from (Berliner, 1884) is included. Also included in each Jastrow entry is a citation of a place where the given entry is used in a Talmudic text. Sefaria’s website links these citations to the corresponding texts found within their database. For example, the citation “Keth. 4a” in Figure 2 links to the Talmud Bavli Ketubot 4a where *Mata Mehasya* is used in the sentence: “Rav Ashi says: It can be found in a place like his city of Mata Meḥasya, which is removed from the category of a city, as it is too small, and removed from the category of a village, as it is too large.” It is important to note that these references are not necessarily comprehensive examples of all the places the word appears in Talmudic texts. It does, however, represent multiple places where the entry is confirmed to be found.

The Brown-Driver-Briggs Dictionary is not found on Sefaria’s website. However, both the BDB and Jastrow Dictionary have been converted into a structured form as part of the MongoDB database. These databases can be explored and queried using MongoDB’s Compass client. All data found within the Sefaria database is available for downloading from Sefaria’s github (Sefaria github). Once downloaded and unzipped, the dump file is uploaded to the Compass client using the command “mongorestore --drop”. This creates a database “Sefaria”, as well as collections within the database of the organized data. I used the collection “lexicon_entry”. This collection holds all entries from all lexicons and dictionaries featured in Sefaria’s database.

Within the Compass client, database information is presented as shown in Figure 3:

[4] https://www.sefaria.org/Jastrow%2C_List_of_Abbreviations?lang=bi

```
{
    "_id": {
        "$oid": "559260bafbfba21d827a02c7"
    },
    "headword": "אֲבַגְתָא",
    "parent_lexicon": "BDB Augmented Strong",
    "content": {
        "morphology": "n-pr-m",
        "senses": [{
            "definition": "Abagtha = \"God-given\""
        }, {
            "definition": "one of the seven eunuchs in the Persian court of Ahasuerus"
        }]
    },
    "strong_number": "5",
    "transliteration": "'Ăbagthâ'",
    "pronunciation": "ab-ag-thaw'",
    "language_code": "x-pn"
}
```


Figure 3: BDB Entry on the Compass Client

Entries can be filtered within the client using search queries. For example, in Figure 3, I specify that I am only searching for entries from the "BDB Augmented Strong" parent_lexicon.

MongoDB is a NoSQL database (Kobielus, 2018), defined as being a non-relational database. A relational database (RDB) is a grouping of data items held in tables. In a relational database each row is one entry, and each column is an attribute of the entry. The uniqueness of each entry is maintained using a unique attribute known as a primary key, for example a unique ID. Information held in different tables are connected using a relational attribute that is common in both tables called a foreign key, an example of this could also be an ID held in both tables. This connection is called a join. Every entry has a space allocated for every attribute, if the attribute information is missing for that entry then it will be presented as a Null or shown to be missing in some other way.

In contrast MongoDB is highly structured in how it holds data, but is less rigid than a RDB. Rather than holding data in the form of tables, each entry is presented in the form of a JSON (JavaScript Object Notation) object. The structure of a MongoDB entry is very similar to an Object, Dictionary or List, in that there is information is held in the form of fields and

values. Information is accessed through indexing into fields and finding the values associated with it. Each value is also able to be a sub dictionary of fields and values.

Figure 3, for example, shows an entry that has the fields and values of a unique ID; a headword that holds the Hebrew word; the parent_lexicon - that points to which lexicon the entry information comes from. The value of content is a dictionary object that holds the values morphology and senses. Content.morphology defines what part of speech the word is, while content.senses holds an array where each element is a dictionary with a key "definition" and the value being one definition of the word. The entry object also holds the fields and values: strong_number - the number given to all biblical words by James Strong in his Concordance of the Christian Bible; the transliteration of the headword to English characters; the transliteration based on pronunciation; and which language the word originates from.

The Jastrow dictionary entries have a slightly different structure to the BDB entries.

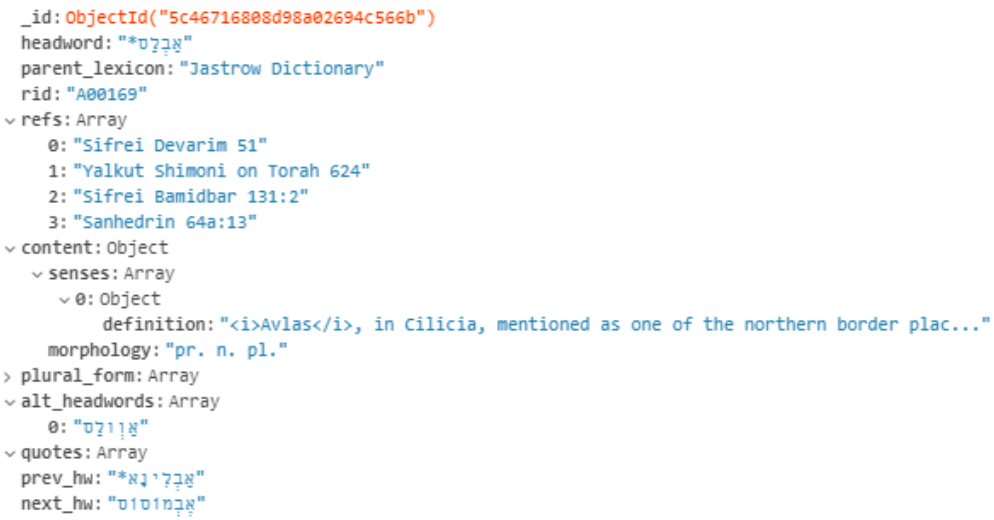

Figure 4: Jastrow Entry on MongoDB

Included in the Jastrow entry are fields such as references - where the word is found, alt_headwords - alternative formulations of the word, as well as prev_hw and next_hw - pointers to the previous and next entry.

MongoDB enables the creation of a rigid schema, where fields are predefined before the creation of the database, and entries must conform to the schema. However, the Sefaria MongoDB does not use utilize this convention. Therefore, each entry may have a completely different structure compared to another entry in the database. For example, as seen in Figure 3 and Figure 4 the fields and structure, as well as where information is held, is vastly different between the two dictionary entries, even though they are both found in the same "lexicon_entry" collection. Furthermore, the same information may be held in multiple places in one entry, such as the part of speech information which may be found in a Jastrow entry under the field content.morphology or content.senses.definition, making the search queries needed to extract information from the entries somewhat convoluted, in an attempt to make sure that all available data is gathered. Even within one kind of entry, such as under all Jastrow entries, not every field will be present.

The gazetteers were created using three separate searches of location names found in Jastrow, and male and female names found in both the Brown-Driver-Briggs and Jastrow dictionary. Although the Brown-Driver-Briggs is a lexicon exclusively for Biblical words, an assumption was made that it would not be unlikely for certain characters in the *Tanach* to mentioned within the Talmudic texts either in direct quotes where the biblical personalities are quoted as saying something or as characters who are referenced and discussed.

Search queries were developed in order to address the difference in structure between entries that originated from either the BDB or Jastrow dictionaries. For example, since entries in the BDB Lexicon are all tagged with a morphology, all BDB entries defined as being a name will have the content.morphology field tagged as either "n-pr-f" or "n-pr-m". While some Jastrow entries do have a content.morphology tag, content.morphology does not always contain the needed information and the morphology is instead found in content.senses.definition. Another issue that is addressed through the search queries is that

there are some Jastrow named entity entries that are given a more general content.morphology tag of “pr. n.”. However, although not all pr. n. entries are named entities, some are in fact a female or male name. In order to gather these entries, extra words are searched for within content.senses.definition such as the word “woman”.

Table 1: Female and Male Names Queries

parent_lexicon	content.morphology	content.senses.definition
BDB Augmented Strong	n-pr-f	
	n-pr-m	
Jastrow Dictionary	pr. n. f.	
		pr. n. f.
	pr. n.	woman
	pr. n. m.	
		pr. n. m.
	pr. n.	Angel Divinity Family Demon Sorcerer Son Deity Tribe Surname Spirit Patriarch

As seen in Table 1, the technique of supplementing morphological tagging with keywords searched for is used extensively with words that might not be a specific "person", but is still a Named Entity, such as a supernatural being.

Additionally, this technique was also extensively used in gathering location entries. While some entries had the content.morphology tag of pr. n. pl.- proper noun of a place, sometimes the tag was found in content.senses.definition and many only had the pr. n. label. Some of the definitions of place names cited as coming from various geographical dictionaries, the abbreviations for these books are included in the content.senses.definition query. These abbreviations include Berl. Beitr. = Berliner Beiträge zur Geographie und Ethnographie Babyloniens, Berlin 1884, Hildesh. Beitr. = Hildesheimer Beiträge zur Geographie Palestinas, Berlin 1886, and Neub. Géogr. = Neubauer Géographie du Talmud, Paris 1868[5]. In order to filter out pr. n. entries that are location names, key terms that might be included in the entry definition, and which may signify that the dictionary entry is probably a location term, are included, such as "mountain", "kingdom" or "city".

Table 2: Location Query

parent_lexicon	content.morphology	content.senses.definition
Jastrow Dictionary	pr. n. pl.	
		pr. n. pl. Berl. Beitr. Hildesh. Beitr. Neub.
	pr. n.	town country district province

[5] ibid

		city canal river tributary street brook valley border mount mountain kingdom land cave lake pond gate tower building peninsula settlement

A cursor object is returned once the find method is called with the search queries implemented above. The cursor points to the documents that match the query. The collection of entries is turned into a list and then iterated over. Both ‘headword’ and ‘alt-headword’ are indexed into and processed to be added to the gazetteer. Using regex all vowels, as well as other non-alphabetical characters, were removed. Many alt-headwords feature apostrophes that signify an incomplete word. Heuristic methods were used to process these cases to create a complete entry. For example, under the entry פאגי is the alt-headword ‘בית פ, the apostrophe is replaced and the complete entry בית פאגי is added to the location gazetteer.

The contents of the location set are written to a gazetteer location text file with the tag LOC, while the contents of the person list are added to another text file with the tag PER.

These text files are used later in the training process by the NER. The gazetteer files are used initially to tag the text. This tagged text is then used to generate other features. These generated features will be used to train the machine learning model which will be the basis for the NER.

A major challenge in training a named entity recognizer for the Talmud is tagging multi-token entities. Multi-token entities are singular named entities which are made up of multiple smaller tokens. A common example in the Talmud would be the name of a *Rav;* for example, רב שמואל בר יצחק - Rav Shmuel Bar Yitzhak is a singular named entity, but has multiple components such as his title, name and his father's name. This issue is dealt with on the tagging level using IOB tagging. Standing for Inside, Outside, Beginning this method of tagging allows multiple words to be represented as being connected to each other and signify one named entity. Tagging the multi-token named entity mentioned in Figure 2 of *Mata Mehasya* would look like this using IOB tagging: ['מתא', 'B-LOC'], ['מחסיא', 'I-LOC'].

Features which are used to train the machine learning model are generated based on various features of the word as well as surrounding words. The features chosen were based on the work of Naama Ben Mordecai and Michael Elhadad who developed a NER system for Hebrew (Elhadad, 2012). The features include information of the word itself such as the prefix and suffix and information about the words before and after. The gazetteers created are also used in generating the features by using the result of looking up the word in the gazetteer.

Table 3: Features Generated

Word Position	Feature	Result Data Type
Word Itself	First in sentence	Bool
	Last in sentence	Bool

	word	string
	prefix	string
	suffix	string
	Found in gazetteer	Bool
Word Before	word	string
	Found in gazetteer	Bool
Word After	word	string
	Found in gazetteer	Bool

The initial tagged data utilized for training is created using the gazetteers to tag words as either location or person. This is not a fully accurate method of tagging as it does not consider all forms of the words that can be found within the text, such as different prefixes or suffixes. Nonetheless, this data is then sent through the features generator and turned into a vector to be used with the Machine Learning model. The data is then split into training and testing groups.

I used the NLTK Naive Bayes Classifier to implement the machine learning portion of my research. A Naive Bayes Classifier uses Bayes' Theorem to calculate the probability of data being classified, assuming the independent and equal contribution of each feature to the overall probability (Naive Bayes Classifiers, 2021). This assumption is naive as is it is likely that the different features are dependent on each other to some degree (ird, 2006). Despite the simple math that the model is based on, as well as the assumptions made by the classifier, Naive Bayes Classifiers have high accuracy in real world use, especially textual classification problems (Naive Bayes).

Equation 1: Bayes Theorem

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$ [6]

Results

As a result of the data extraction done on Sefaria's MongoDB two gazetteers were created. The gazetteer that holds all location names gathered from the Jastrow Dictionary has 1689 entries. The gazetteer of people names gathered from the Jastrow and BDB dictionary has 2676 entries. A scan of the results shows that while most words inserted into the gazetteers as either a person or location are indeed a Named Entity, there are instances of noisy data. For example, in the person gazetteer an entry "מצפה" is included, a word that refers to a place but not a person.

Future Work

There are many avenues open, for where future work can be done in order to improve on the research already done. Although the process of gathering data and machine learning, as described above, has shown promising results, there are improvements that could be made on the tagging of texts used for training, as well as on other aspects of the training.

My current process leverages gazetteers created from a few sources, the Jastrow and BDB dictionaries, and Satlow and Sperling's database of Rabbinic names. These gazetteers are then utilized in an initial tagging of the text that is then used for training. There are multiple issues with this technique of text tagging: that the gazetteer will tag words which are

[6] https://en.wikipedia.org/wiki/Bayes%27_theorem

not a named entity, but merely a homograph of that word, or that it will not tag words that are named entities as the word has prefixes and suffixes so is not an exact match.

One potential source for a more accurate tagging of the text is from the various examples of texts that gather people's names in the Talmud and include citations for places in the Talmud where those people are mentioned in the text. For example, Jastrow will include multiple citations for places in the Talmud where each entry can be found. Citations can also be found in *Oẓar Divrei Ḥakhamim u-Fitgameihem,* where each saying is attributed to a Rabbi. Therefore, these citations could potentially be used as a way of accurately tagging the Named Entity associated with either the entry, or the rabbinic saying.

This strategy has its downside as the citations included in the Jastrow are not inclusive of all places where the word can be found in the texts. However, one positive from using this strategy is that all examples are confirmed examples of the Named Entity being used in the text. Therefore, it is potentially a way of creating an almost hand tagged text. Another issue that comes from potentially using one of the rabbinic bibliographic books is that these books only show examples of Rabbinic sayings, and don't include examples of saying by other kinds of non-rabbinic people.

Using a lookup method for tagging as well as generating features may require some flexibility. Only using exact matches is likely to miss many places where the word has prefixes, suffixes or some other kind of grammatical form. Techniques available to allow for a less rigid use of the gazetteers include stemming or lemmatizing and fuzzy matching (Nadeau, 2007). Stemming and lemmatization are techniques of word processing that shorten or standardize different formats of a word down to a reduced version. This reduced version gets rid of the different disparate formats of the word so that it is more likely to be able to be matched to the gazetteer form of the word. Another method for enabling more extensive tagging is through fuzzy matching, this will match words which have a small edit distance

and are therefore very similar.

Further developments of the current process would be to attempt the machine learning portion using different models. The “naive” nature of the Naive Bayes model basing its classification on the assumption of independence of the many features can sometimes be detrimental in the effectiveness of the model. Furthermore, the current implementation of the model mainly generates features on the word itself and uses very little information taken from surrounding words.

This is a particularly relevant issue to solving the goal of NER on Talmudic texts, as many names found in the Talmud are of those of Rabbis who often have many parts of their name, such as their title, name and their father’s name. Being able to take into consideration the context is helpful for a model being able to tag the entire name as a Named Entity. Therefore, using other models instead that consider the context of the entire sentence, as well as not treating the features as independent may produce improved results. Some examples of such models can be seen in projects discussed previously which also use ML models to solve NLP problems associated with the Hebrew language.

One example of such a model would be to use a Conditional Random Field (CRF) based NER. CRFs used on NER problems will calculate the probability of the current word being labeled as a Named Entity given the context of the whole sentence. To do this the sentence is modeled as a graph, where interdependencies between the different words are implemented[7].

Conclusion

Despite the work that can theoretically be done in order to improve the research done so far, as described above, much progress has been made to achieve the goal of creating a Named

[7] https://en.wikipedia.org/wiki/Conditional_random_field

Entity Recognizer for Talmudic texts. The gathering of named entities from a variety of sources and formatting those words into gazetteers is something that is extremely valuable in the NER process. What is particularly noteworthy is that much of the previous work on NER and Talmudic texts have focused mostly on rabbinic figures, leaving out valuable information on Named Entities such as location names, and other person names who are not rabbinic.

Another valuable aspect of this work is the survey done on multiple different venues researchers in the Hebrew NLP field have taken. An important aspect of this survey is that although work might be done in order to improve NLP using a technique, that process, or tool created might need adaptation in order to use on Hebrew texts from different time periods. The Hebrew NLP field is already small, and different research teams will face many of the same challenges. However, despite the overlap, improved accuracy may be achieved through training on texts specific to the time period that the research is aimed at addressing.

Works Cited

Avi-Yonah, M. (1966). *Carta's Atlas of the Period of the Second Temple, the Mishnah and the Talmud.* Israel: Carta.

Berliner, A. (1884). *Beitrage zur geographie und ethnographie Babyloniens im Talmud und Midrasch.* Berlin: J. Gorzelanczyk & Co.

Bonayich . (2021). Retrieved from Otzar Hadmuyot: https://www.bonayich.com/project/otzar-hadmuyot/

Brown, F. S. (1996). *The Brown, Driver, Briggs Hebrew and English Lexicon : with an Appendix Containing the Biblical Aramaic : Coded with the Numbering System from Strong's Exhaustive Concordance of the Bible.* Peabody, Mass: Hendrickson Publishers.

Elhadad, N. B. (2012). Hebrew Named Entity Recognition.

Heilprin, J. b. (1961). *Sefer Seder Ha-Dorot : Sheloshah Sefarim Niftaḥim Bo : 1. Seder Hadorot, 2. Seder Tana'im Ṿe-Amora'im, 3. Shemot Ha-Meḥaberim Ṿeha-Sefarim.* Hotsa'at Leṿin-Epshṭayn.

Hyman, A. (1902). *Sefer Bet Ṿa'ad La-Ḥakhamim.* London: ḥ. mo. l.

Hyman, A. (1947). *Otsar Divre Ḥakhamim U-Pitgamehem.* Tel-Aviv: Hotsa'at "Devir".

Hyman, A. (1964). *Sefer Toldot Tana'im Ṿe-Amora'im.* Yerushalayim: Hotsa'at Ḳiryah ne'emanah.

ird, S. E. (2006). Learning to Classify Text. In *Natural Language Processing with Python.* Sebastopol, CA: O'Reilly Media.

Jastrow, M. (1926). *Dictionary of the Targumim, Talmud Bavli, Talmud Yerushalmi, and Midrashic literature.* New York: Choreb.

Kareem Darwish, N. H.-K.-N.-S.-B.-H. (2021, April). A panoramic survey of natural language processing in the Arab world. *Communications of the ACM, Volume 64, Issue 4*, pp. 72-81.

Klein, R. T. (2019). What's Wrong with Hebrew NLP? And How to Make it Right. *CoRR*.

Kobielus, J. (2018, June 27). *MongoDB Drives NoSQL More Deeply into Enterprise Opportunities.* Retrieved from wikibon: https://wikibon.com/mongodb-drives-nosql-deeply-enterprise-opportunities/

Kosior, W. (2021). "Six Things Are Said Concerning Demons" (Hagigah 16a). The System of Topic Tags Used in the Elyonim veTachtonim Inventory to Describe the Features of Supernatural Entities and Their Relationships with Humans. *The Polish Journal of the Arts and Culture. New Series 13*, 109–131.

Margaliot (Margulies), Mordecai . (2021, Nov 25). Retrieved from Encyclopaedia Judaica.: https://www.encyclopedia.com

Margalioth, M. (1945). *Entsiḳlopedyah Le-Ḥakhme Ha-Talmud Ṿeha-Ge'onim.* Tel-Aviv: Y. Ts'ets'iḳ.

Nadeau, D. a. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes 30.1*, 3-26.

Naive Bayes. (n.d.). Retrieved from scikit-learn: https://scikit-learn.org/stable/modules/naive_bayes.html

Naive Bayes Classifiers. (2021, December 12). Retrieved from geeksforgeeks: https://www.geeksforgeeks.org/naive-bayes-classifiers/

Ne'eman, P. (1970). *Entsiḳlopedyah le-ge'ografyah Talmudit : be-tseruf mapot tsilumim ṿe-luḥot.* Tel Aviv: Y. Ts'ets'iḳ.

Neubauer, A. (1868). *La Géographie du Talmud.* Germany: Lévy.

Neusner, J. (1990). *Language as Taxonomy: The Rules for Using Hebrew and Aramaic in the Babylonian Talmud.* Atlanta: Scholars Press.

Parker, J. (2005). *Sages of the Talmud*. Retrieved from https://web.archive.org/web/20190129045302/http:/www.joshua-parker.net/sages/.

Rusinek, S. a. (2021). Feeding a Gazetteer: Leveraging Word Embeddings for Toponym Mining. *roceedings of the 5th ACM SIGSPATIAL International Workshop on Geospatial Humanities*, 28–35.

Sefaria github. (n.d.). Retrieved from Sefaria-Export: https://github.com/Sefaria/Sefaria-Export#readme

Sherira ben Hanina, G. a. (1988). *The Iggeres of Rav Sherira Gaon.* Ahavath Torah Institute-Moznaim.

Shmidman, A. a. (2020). A Novel Challenge Set for Hebrew Morphological Disambiguation and Diacritics Restoration. *Findings of the Association for Computational Linguistics: EMNLP 2020*, 3316–3326.

Shmidman, A. S. (2020). Nakdan: Professional Hebrew Diacritizer. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 197–203.

Solomon Schechter, W. B. (n.d.). *JewishEncyclopedia*. Retrieved from MATAH MEḤASYA (MAḤSEYA): https://jewishencyclopedia.com/articles/10473-matah-mehasya-mahseya

Sperling, M. S. (n.d.). The Rabbinic Citation Network. *forthcoming in AJS Review*.

Waxman, J. (2021). A Graph Database of Scholastic Relationships in the Babylonian Talmud. *Digital Scholarship in the Humanities, Volume 36, Issue Supplement_2*, ii277–ii289.

Zhitomirsky-Geffet, M. a. (2018). SageBook: Towards a Cross-Generational Social Network for the Jewish Sages' Prosopography. *Digital Scholarship in the Humanities (DSH)* .

Zinberg, I. a. (1975). *The German-Polish Cultural Center.* Hebrew Union College Press.