The goal of this document is to provide a list of features for the Vocabulary Manager project which have not yet been implemented but which should be eventually developed to make the project more comprehensive and original. Each feature listed will include a description of the feature, why the feature is useful, and some suggestions for how the feature can be implemented. The features are listed according to their current perceived utility to the project, with higher utility features being presented first.

Features

Database Queries

Once users mark words in a text as new vocabulary and save the words to the Vocabulary Manager database, they should have some method of retrieving these words so that they can form vocabulary lists. With this in mind, the user should be able to issue database queries to build vocabulary lists. The method of issuing queries should be controlled, however. What this means from a technical standpoint is that the user should not be able to issue MongoDB queries directly, but instead should have some interface for building queries which can then be converted into MongoDB queries through back-end code.

Such an interface should, at minimum, allow the user to build different filters which the user can then combine through basic Boolean operators like AND, OR, and NOT. The filters themselves should filter on attributes such as title, author, work, page, and date the word was added. More advanced filtering criteria can be proposed later, but this should serve as a sufficient starting point. The overall goal is to provide an interface that will allow the user to build sophisticated queries that lead to comprehensive vocabulary lists while simultaneously not exposing the full MongoDB query language, which could lead to security issues as well as require too much technical training from the user.

The results from a database query can be returned to the user in one of two ways. Either the full list of vocabulary words and their translations can be returned, or a list containing vocabulary words with their translations hidden can be returned. In the latter case, users could then click on a vocabulary word to see the translation. This workflow mimics the study aid of a folded piece of paper with vocabulary words on one side and translations on the other. Regardless of how the vocabulary words are originally displayed, query results can also form the basis for vocabulary retention tests, which are discussed in a later section of this document.

Users should also be able to have options for managing their queries. They should be able to save their queries for future use, see how the results of queries change over time, and see statistics for how queries are used.

External Vocabularies

This feature will enable users to consolidate vocabulary that is acquired from external sources that Vocabulary Manager doesn't directly interact with, such as newspapers and magazines. It should allow users to upload vocabulary words along with information about their sources. Once the vocabulary words are uploaded and saved in a database, users can then query them in the same manner as vocabulary words acquired through the document viewer or similar tools.

There should be at least two different types of interfaces for uploading external vocabularies. The first is a simpler interface for uploading a few words. This interface should prompt users for vocabulary words to enter and then prompt them for information about sources. Users can then upload vocabulary words at the end of their interaction with the interface. The second interface can be driven by interactions with external files. In particular, users can upload external files containing a larger set of vocabulary words, and this set can then be uploaded to the database. This option will be useful when the user doesn't want to spend too much time interacting with a GUI to upload a large amount of words. The challenge here, however, will be to come up with a data exchange format to support automatic vocabulary uploads, but this shouldn't be too difficult. The format can most likely be based supported through JSON.

Website Parsing

Since websites can contain lots of text, users should be able to read the text from them in Vocabulary Manager. Thus, there's a need for functionality to parse HTML from websites and convert the appropriate text into a document that the user can then interact with through the document viewer. To accomplish this, the user should be able to enter a URL, at which point Vocabulary Manager will fetch the HTML from the website, parse it, and convert it into a document.

A first version of this feature can be implemented without user modifications before the document is saved. In particular, only meaningful text should be extracted from a website's HTML and no additional effort should be made to try and render pictures or other graphics from the original site. Although this will appear to be a reduced version of the original website, if the user is primarily interested in reading the text from the site, this shouldn't be much of an issue.

Another technical consideration that is important to mention is that if the text the user wants to read is actually split across multiple similar web pages from the same domain, as can be the case when reading works in the public domain, there should be support for specifying how such web pages should ultimately be parsed, either through user input in a GUI or through third-party plug-ins.

Subsequent versions of this feature can request feedback from the user before a document is uploaded. Perhaps the most useful feedback would be determining where page boundaries should be, but users could have other options as well, such as including some images or simple graphics in order to provide a more pleasant aesthetic that somewhat recreates the original web page.

Vocabulary Recommendation

This feature will scan through a user-provided document to find foreign-language words that the user most likely doesn't know. Once those words are discovered, they will be presented to the user along with their corresponding translations so that the user can memorize them before reading the provided document. The benefit of memorizing potentially unknown words before reading is that the user won't be constantly interrupted by looking up new words during the reading process, which will in turn allow the user to instead focus on higher-level interpretations of the document, similar to reading in his native language.

Since this feature appears to be more novel, its implementation will require more research. At the very least, NLP techniques such as text classification and topic discovery will have to be leveraged in order to guess the semantic context of a user-supplied text. This semantic context can then be compared with the estimated vocabulary of the user in order to determine which words in the document will likely be of interest. Beyond this, it will also be useful if the vocabulary recommendation algorithms can learn from user feedback, so it may be of worthwhile to explore Reinforcement Learning for this purpose.

In any case, it may not be too difficult to get a prototype of this feature working, but it's likely that NLP and Machine Learning techniques will be necessary for more sophisticated vocabulary inference in the future.

Third-party Dictionary Services

There may be cases where a user needs to use a different foreign-language dictionary than the one provided by Vocabulary Manager. Such use cases will most likely occur when a user is trying to interpret a text that is dealing with a specialized, possibly technical domain. For example, when I was working on processing German-language medical documents at a previous job, I had to constantly refer to specialized German dictionaries that explained medical terminology and abbreviations.

The question then is how can this be integrated with the Vocabulary Manager project? For the moment, I could imagine third-party dictionary providers offering REST APIs that Vocabulary Manager can call for a user during the dictionary lookup process. The user could then be charged separately by the third-party for his use of the vocabulary service, if applicable. To use third-party APIs, however, means that that APIs will have to conform to a standardized interface as specified by Vocabulary Manager. It should be possible to derive a suitable interface without customer feedback for now, but eventually it would be necessary to get vendor input.

One drawback to using third-party dictionary services is that there will be extra latency in returning results from the Vocabulary Manager dictionary service because third-party services will have to be called across the network. It may be possible to at least partially get around this by caching common queries to those services and their results, but the user should be aware that he might have to pay a performance penalty the more dictionaries he consults.

Advanced Vocabulary Retention Tests

One of the main benefits of having a consolidated vocabulary tool like Vocabulary Manager is that a user can test how well he retains new vocabulary over time. I've seen similar services use flashcards for testing retention, but it's possible to go beyond this. To start with, one could test the user on inflected forms of the vocabulary he's acquired as well as on grammatical information like a word's part-of-speech. This will aid in a more rapid acquisition of the morphology of the language being learned.

We can take this a step further and test for syntax and semantics as well. For syntax, we could first estimate how well a user can process different syntactic complexities of the language he's learning and based on this we can then have the user try to translate sentences of similar syntactic complexity which contain similar words to the user's recently acquired vocabulary. For semantics, we could try to figure out topics the user's recently acquired vocabulary falls under, and then test the user on other vocabulary items related to those topics. Furthermore, we may also test the user on recognizing when foreign-language words are synonyms or antonyms, to facilitate semantic comprehension.

Vocabulary Retention Statistics

After vocabulary retention tests have been completed, it would be beneficial to see statistics comparing current test results to previous test performance. This would help to quantify how effective a user is at retaining new vocabulary. Furthermore, Vocabulary Manager could eventually expose these statistics to verified third-party vendors through an API so that they could develop more individualized learning plans for users. Vocabulary Manager could also try to automate some of this tutoring process based on test results, but this may be too involved for the immediate future.

From a UI perspective, statistics should be displayed to the user through a graph or tabular format. The user should also be able to choose which statistics he does or doesn't want to see, and he should be able to generate meaningful reports automatically. This will enable the user to make more robust, data-driven decisions when deciding what to study next.

As for what the statistics themselves will ultimately be, there are many options and they can be more thoroughly elaborated later. To start, however, we could see how effective a user is at retaining words in different part-of-speech classes, words from different authors/topics, and words that are complex either due to morphological features such as compounding or empirical factors such as low occurrence in text corpora.

In any case, the statistics chosen should be comprehensive and allow the user to make data-driven decisions, but they should not be too overwhelming either, so more effort will be required here to find the right balance.

Optical Character Recognition (OCR) Support

This feature will allow users to upload a scanned document containing text from a foreign language. Once the document is uploaded, it can be run through an OCR algorithm and the corresponding text will be displayed to the user. At this point, the user can then modify any incorrect text, save the resulting document, and then interact with it in the document viewer just like any other text.

At first glance, this feature may not seem particularly useful, since lots of texts are already digital these days anyways. The feature does provide some utility upon closer inspection, however. First, some texts like newspapers and magazines don't always have corresponding digital editions. If a user feels like he needs a fair amount of dictionary support to look up words while reading such texts, OCR support will enable this. Second, the user may be trying to read an old book, which has no digital equivalent. OCR support will once again make it easier to convert the book into a format the enables dictionary lookup.

Overall, this feature may end up being applicable to a smaller set of users compared to other features suggested in this document, but it should still provide enough utility to be worth the effort to implement it.

Audio Processing

Although the original intent of Vocabulary Manager was to work with text data, there are also benefits to working with audio data. In particular, being able to process spoken language will assist users in being able to speak and listen to a foreign language, which will in turn allow them to go beyond just written comprehension. To this end, Vocabulary Manager should support the processing of audio data.

Support for this type of processing should start with automatically transcribing audio data by producing the corresponding words in the audio clips along with the time in the recording at which each word occurs. Similar to the workflow for OCR support, users can then modify any potential mistakes with transcribed words and then convert the resulting data into a document that can be viewed with the document viewer.

Since we're dealing with audio data, we can also save individual words as audio clips and use these clips for vocabulary retention tests later on. We could also try processing words spoken by users, but we may run into issues here as speech-to-text models are most likely not trained to handle input from non-native speakers very well.

Finally, from an implementation standpoint, the CMUSphinx project seems like it may be a good place to start to support audio processing, but other speech recognition packages should be researched as well.

Semantics

There are multiple opportunities for integrating semantic content in order to make the application more intelligent. At a basic level, there should be support for users to bookmark pages, add pages to a table of contents, and insert page notes. These features, while comparatively simple, will allow the user to personalize the text he is reading, which will ultimately enable deeper understanding.

One does not need to stop here, however. A more interesting feature would be to rank translations of a user's query based on local and global semantic content of the text he is reading. For example, the German word *Anschauung* usually has a precise technical translation as "intuition" when reading the works of the philosopher Immanuel Kant, but it can also be translated more loosely as "view" or "opinion" in other contexts, such as in the German expression *Weltanschauung* (worldview). From a more practical standpoint, this feature would spare users from the effort required to read through translation results that are not necessarily as relevant, which is one of the shortcomings of using online foreign-language dictionaries to look up unknown words or phrases. We can generalize this idea even further and provide users with tools for configuring and customizing their translation results, such as prioritizing one type of part-of-speech over another, or preferring translations that are composed of more common words in the target language.

In a similar fashion, we could also provide other semantic services, which return synonyms or antonyms of the text to be translated along with the corresponding translations. In this way the user can then build up his vocabulary even more quickly by acquiring multiple vocabulary words from a single translation request.

Finally, users should be able to integrate ontology content in order to perform reasoning on their documents. For example, if the texts the user is reading mention different philosophers, and such references are correlated with the appropriate ontology entries, then the Vocabulary Manager should be able to deduce insights, with the help of automated theorem proving technologies, such as which philosophers are European or American, and group the texts accordingly. This feature, while potentially requiring more technical sophistication on the part of the user, should also enable a much more meaningful categorization of texts. This could be particularly useful for researchers from academic or scientific disciplines who may have to read journal articles in multiple languages.

Social Media

The features proposed so far do not really enable interactions with other users. This is where support for social media will prove to be useful. In particular, the inclusion of social media capabilities should be primarily focused on allowing learners of different languages to be able to connect with each other based on similar abilities and interests.

To this end, there are two core features that are immediately useful, though others may emerge over time. First, there should be support for user profiles that are accessible by other users of the application. The exact details of what a user profile should contain can be worked out later, but profiles should at least include languages a user is interested in learning, texts read so far in those languages, and goals for future learning. Behind the scenes, recommendation algorithms can be used to find profiles to recommend to a user based on the similarity of texts a user has interacted with through the application, as well as learning patterns derived from those interactions.

Second, users should have the ability to form study groups. Within these groups, users can discuss texts they are reading as well as suggestions for future study. Multiple auxiliary features could also be derived to support this core functionality, such as tools for organizing meeting notes, tools for assigning reading schedules, and tools for suggesting further reading. Furthermore, groups could also set up translation memories that would be accessible to users of that group when looking up words in new texts that they're reading.

Beyond the features suggested here, more research could be done into different existing social media sites in order to generate ideas for new features that will resonate with users. Still, the features suggested here will form a good starting point for enabling like-minded users to connect with each other and practice their foreign-language skills together.