

Capstone

Western Governors University



By: Nathan Pepin

Student ID: 001195100

Contents

[Section A - Nontechnical Proposal 3](#_Toc87554033)

[Letter of Transmittal 3](#_Toc87554034)

[Project Recommendations 5](#_Toc87554035)

[Problem Summary 5](#_Toc87554036)

[Application Benefits 5](#_Toc87554037)

[Application Description 6](#_Toc87554038)

[Object and Hypotheses 6](#_Toc87554039)

[Methodology 6](#_Toc87554040)

[Funding Requirements 7](#_Toc87554041)

[Stockholders Impact 7](#_Toc87554042)

[Data Precautions 8](#_Toc87554043)

[Data Expertise 8](#_Toc87554044)

[Section B - Technical Proposal 9](#_Toc87554045)

[Problem Statement 9](#_Toc87554046)

[Customer Summary 9](#_Toc87554047)

[Existing System Analysis 9](#_Toc87554048)

[Data 9](#_Toc87554049)

[Project Methodology 9](#_Toc87554050)

[Project Outcomes 9](#_Toc87554051)

[Implementation Plan 9](#_Toc87554052)

[Evaluation Plan 9](#_Toc87554053)

[Resources and Costs 9](#_Toc87554054)

[Programming Environment 9](#_Toc87554055)

[Environment Costs 9](#_Toc87554056)

[Human Resource Requirements 9](#_Toc87554057)

[Timeline and Milestones 9](#_Toc87554058)

[Section C - Implementation Report 10](#_Toc87554059)

[Data Methods 10](#_Toc87554060)

[Datasets 10](#_Toc87554061)

[Analytics 10](#_Toc87554062)

[Data Cleaning 10](#_Toc87554063)

[Real-Time Queries 10](#_Toc87554064)

[Adaptive Element 10](#_Toc87554065)

[Outcome Accuracy 10](#_Toc87554066)

[Security Measures 10](#_Toc87554067)

[Product Health Monitoring 10](#_Toc87554068)

[Dashboard 10](#_Toc87554069)

[Section D - Post Implementation Report 11](#_Toc87554070)

[Project Purpose 11](#_Toc87554071)

[Datasets 11](#_Toc87554072)

[Data Product Code 11](#_Toc87554073)

[Hypothesis Verification 11](#_Toc87554074)

[Effective Visualizations and Reporting 11](#_Toc87554075)

[Accuracy Analysis 11](#_Toc87554076)

[Application Testing 11](#_Toc87554077)

[Application Files 11](#_Toc87554078)

[User’s Guide 11](#_Toc87554079)

[Summation of Learning Experience 11](#_Toc87554080)

[Section E- Sources 12](#_Toc87554081)

[Section F - Professional Communication 13](#_Toc87554082)

# Section A - Nontechnical Proposal

## Letter of Transmittal

11/25/2021

Mr. John Smith

WordMem

42 Wallaby Way

Portsmouth, NH 03801

Greetings,

As our company grows and expands to compete with its competitors, it is important to keep pace with emerging technologies. Our current platform only keeps pace with our competitors, such as Memrise and Duolingo, and we are often perceived as being an alternative. Worse, retention rates are low among users who have been using the application for six or more months.

The project proposal recommended by the software engineering team is a machine learning is to implement a text analyzer application that allows a user to input a text document of their choice and to give the user a vocabulary preparation plan.

The primary benefit to the project is that it allows a user to eliminate the overhead of cross referencing. If someone is trying to read a text and they are bombarded with words they have not seen, they will spend much of their time going back and forth between a dictionary. This consequently decreases comprehension and provides a barrier to learning.

WordMem’s basic vocabulary courses prove effective for those who are only starting out learning, but users who gain a basic competency will tend to seek out learning elsewhere to suit their needs. It is common for courses to not align with a user’s interests. By giving these users the tools to direct their learning efforts, user retention can be increased significantly The project goal would be an increase of 10% over the course of a three year period.

The project costs for a basic implementation are estimated at $85,000 with the primary cost being human resources. The estimated upkeep and maintenance are estimated at $15,000.

The initial scope of the project is limited to limit cost and display a proof of value, but if successful, future implementations could be designed to interface with existing WordMem functionality.

If there are any questions, please do not hesitant to react out. Looking to hear from you soon.

Sincerely

Nathan Pepin

Nathan Pepin

Director of Software Development

## Project Recommendations

### Problem Summary

The language learning industry is expanding quickly, and it is easy to fall behind. Currently WordMem’s primary product offering is commonly referred to as a Duolingo clone. WordMem’s prebuilt language learning courses have been proven to be effective, in exit surveys users typically report that their termination was due to course content not being relevant to their interests.

There exist at least two reasons for this issue. The first is that any language course is likely to cover similar materials in the beginning, making it like other courses. The second is that as language learner becomes more competent, that it becomes difficult to create a one size that fits all course.

The solution addressed in this project intends to address both points. The project solution, codename *Text Learner++*, will use cutting edge machine learning to allow users to direct their own learning. It does this by analyzing texts that the user self-selects and developing a vocabulary plan to prepare the reader for reading the text.

### Application Benefits

There are many potential benefits to the application, but the two that are most relevant to the domain of the problem are as follows:

#### Product Differentiation

The project will increase differentiation from alright existing products by allowing a user to determine their own learning path. This provides a much-needed contract against product competitors.

The use of machine learning also enables effective marketing. At the current moment machine learning has captured the interest of the general public and its potential offerings can attract eyes. The solution as proposed in this project will be easy to market to users

#### Self-Directed Learning

Self-directed learning solves many problems simultaneously. Creating an course for a user who knows nothing of a target is relatively easy, but creating a course for a user with a decent understanding of the language is difficult as it is difficult to know what they aim to learn.

People learn languages for many different reasons. John may learn Spanish because some of the sub-contractors at his job speak Spanish and he would like to ease the communication barrier. Elaina may wish to learn Spanish because her girlfriend’s primary language is Spanish, and she is looking for a deeper connection. Earl may be studying Spanish because he got hooked on an addictive Columbian novella series.

In all these instances above the users do not want to learn Spanish for the sake of learning Spanish, but they are learning it for some specific use case. By allowing an user to generate learning material based on their use case, they can more easily achieve their end goal

### Application Description

The application will allow users to text documents, such as books or news articles, and it will a analyze the text using a natural language processing algorithm to determine what are the best words to learn. The application will also allow for some additional information to be presented, such as word frequency and sentiment analysis.

### Data Description

The application will use text documents that the user inputs into the application. The text could be an article or a book, or anything that is in the target language. During processing, the algorithm will remove punctuation such as “!” and stop words such as “then”.

Given that input text could potentially be copywrite protected, the input data is never stored within the company servers, only the extracted words are stored.

### Object and Hypotheses

The primary objective is to improve domain specific vocabulary comprehension by allowing a user to directly input the domain specific source and outputting the most important words in that source.

A domain in this instance is a set of vocabulary that is specific to a specific context. The language domain of medicine is likely to contain many words describing anatomy and symptom lists. The domain of construction is likely to be much different and contain many words about lumber, power tools, and layouts.

The hypothesis is that presenting user’s domain specific vocabulary that their domain specific knowledge will increase.

A following hypothesis is that if a user studies for the most important words for a text prior to reading said text, that their comprehension will be increased greatly by decreasing the overhead of cross referencing. Every time that a person needs to identify the meaning of a word when reading a text is a moment where a user drops the flow and context of what they were reading, which when done too much can incite confusion and frustration.

### Methodology

The initial implementation to create the project will be a waterfall broken down into the following.

1. Requirement gathering: Find all possible requirements for the project, not limited to:
   1. Database implementation
   2. Program language
   3. Potential API hooks
   4. Legal concerns
2. Design: Draft program architecture and functionality.
3. Implementation and testing:
   1. Put design plan into action.
   2. Test results on sample data to assess functionality.

After the implementation phase, the further phases will use a more agile approach. If a user in the closed beta finds a bug or suggests a great feature, a ticket will be generated and go through a similar process to 1-3.

1. Closed beta
   1. Restricted to VIP personal.
2. Open beta
   1. Listed as an experimental feature for standard users.
3. Full release
4. Maintenance
   1. Optimize algorithm.
   2. Fix bugs.
   3. Address user feedback.

The purpose of the scaled release is to have the ability to fine tune the algorithm based on user feedback before going live. This is a concern because machine learning algorithms are known to work well in test environments, but to fail when exposed to a large user base, which can create a bad impression. By scaling the release, user expectation can be managed more effectively.

### Funding Requirements

The project funding is estimated at $85,000 being human resources. A full launch is expected to be after a year of development.

The estimated upkeep and maintenance are estimated at $15,000 per year with the main cost being server and computer resources.

Resources will be used to different degrees at different points of the project. The initial phase for instance will utilize two programmers, whereas the project will shift to one part time programmer upon entering closed beta.

The cost breakdown is as follows:

* Programmer: $5,000/month
* Server resources: $200/month
* Compute Resources: $600/month

### Stockholders Impact

The internal stakeholder impact, beyond potential increases in revenue, would affecting the company in the following way

* Marketing would create targeted advertisements to draw in users via the new application.
* IT and database administrators would be required to the host project server and database.
* Customer service would be required to deliver support for the product when in full release.

The project would have the largest impact on users, who would have a new tool to continue their language learning journey with an overall language comprehension increase.

### Data Precautions

The primary precaution in this project is that the input data is potentially copyright material. For the processing outputs this will not be an issue as individual words are not subject to copyright, but if copy right material is inserted into the database it could become a liability.

Given that text will be inserted into a database, it is important to be weary of potential SQL injection attacks. Though the probability of this is low due to words being parsed in a delimited fashion, it is still important to prevent any potential vulnerabilities by using input variables in and SQL code.

Another precaution is regarding some data sources potentially being expensive and unproductive in analyzing. If someone for instance inputs a 1GB text file of their pet monkey smashing on the keyboard, although certain parts could possibly be profound, the result of processing the file is likely to be a waste. Thus, the project will set out guidelines to prevent pointless or malicious computations.

### Data Expertise

The developers that work on the project must have at least three years of programming experience and be familiar natural language processing. A solid understanding of data structures and computational complexity are key. Though optional, developers with some learning language experience would be preferred.

# Section B - Technical Proposal

## Problem Statement

Users who use WordMem’s language learning application lose interest in the company’s primary product as they gain more proficiency because the course material is not directed towards their aim. A course that teaches basic vocabulary such as “hi” and “cat” are effective for most all beginners and will meet with high engagement, but a course that teaches specialty words outside the learner’s target domain is likely to be met with low engagement as it may not meet their needs.

The problem that this proposal attempts to resolve is that of self-directed user learning, where a user can provide to domain of their language that they wish to increase their proficiency in. The proposal will refer to this as *domain specific language*, clearly defined as:

*Domain specific language: Words that loosely correlate with a specific topic.*

## Customer Summary

The program will allow users to input self-selected texts that are relevant to their domain. If a user wants to learn standard business terms, they may input some business books. If they are attempting to learn about music, they may input a book on music. If they want to read a great novel, they may input the classic tale of Moby Dick.

The input will be processed through a natural language processing algorithm and determine what words are most important in the input text and display them in rank order with their frequency. For instance, in the case of Moby Dick it may display:

('whale', 1032),

('one', 906),

('like', 561),

('upon', 541),

('ship', 539),

('man', 481),

('sea', 445),

('ye', 434),

('old', 431),

('boat', 429),

('time', 429),

The user will parse through the words and add words that they do not already know to their learning list. The user may already know half of the words above but may have not seen the word “whale” or “ship”.

The user can use the information in two manners. They can use it as a prelude before reading the book. This can aid greatly in reducing cross referencing and increasing general comprehension.

The second is that a user can use it to extract domain specific language. A user for instance may input several books on carpentry that they have no intention of reading to get a sense of what words are used in that domain.

The user will be given the ability to query the distribution of the top words so that they can understand what words are used where. In technical texts, it is often the case that a word may only be important in certain chapters, meaning that a user could carefully select what words to learn in order to make their learning efficient.

Further, the user will be given a overall sentiment analysis that will inform them of the general sentiment of the tone. This could be helpful in a screening process. If a user is looking for an uplifting text and the sentiment scores it as being very negative, it may prevent the user from wasting their time.

## Existing System Analysis

The primary application that WordMem offers its customers is a collection of spaced repetition vocabulary courses that are designed to structure language learning. The application will start with basic greetings and process to common animals and other everyday topics.

The initial project proposal intends to offer an application in tandem with the current application. The proposal will utilize:

* SQLite database connection
* Python 3.9.5 for data processing
* Jupyter Notebook for UI
* NLTK package for language processing

## Data

There are two primary datasets that will be used, the natural language dataset which contains language specific information, and the text inputted by the user.

### Input Text

The user will input a self-selected text that they intend to learn from. The file must be in a .txt format. It is important to be careful to not store the raw input data on company serves as it may be subject to copyright.

### Natural Language Processing

The application will use a free open-source library called NLTK that has pretrained language models which enables for data cleanup and processing. The text will be into a list of words, punctuation such as “!” will be removed, stop words such as “then” and “by” will be removed as they are generally not important.

The words will then be reduced into their base meaning and counting, counting different forms and conjugations as a single instance. For instance, the word “mice” and “mouse” would be flattened to just “mouse”.

## Project Methodology

The initial implementation to create the project will be a waterfall followed by an agile workflow. The primary objective of the waterfall phase is to build a basic implementation of the deliverables (as detailed in Project Outcomes). The proceeding agile phase will use a scaled release and intend to refine the application according to user feedback.

The purpose of the scaled release is to have the ability to fine tune the algorithm based on user feedback before going live to a wise audience. This is a concern because machine learning algorithms are known to work well in test environments, but to fail when exposed to a large user base, which can create a bad impression. By scaling the release, user expectation can be managed more effectively.

## Project Outcomes

Upon project completion, WordMem customers will have the ability to input texts that they wish to learn from using a graphical user interface. The application will output them the top words to learn in additional to some helpful views.

## Implementation Plan

The application will be hosted on an internal developer and production environment, both hosted on Azure. The following provides a rough outline of the project implementation.

1. Requirement gathering: Find all possible requirements for the project, not limited to:
   1. Database implementation
   2. Program language
   3. Potential API hooks
   4. Legal concerns
2. Design: Draft program architecture and functionality.
3. Implementation and testing:
   1. Put design plan into action.
   2. Test results on sample data to assess functionality.

After the implementation phase, the further phases will use a more agile approach. If a user in the closed beta finds a bug or suggests a great feature, a ticket will be generated and go through a similar process to 1-3.

1. Closed beta
   1. Restricted to VIP personal.
2. Open beta
   1. Listed as an experimental feature for standard users.
3. Full release
4. Maintenance
   1. Optimize algorithm.
   2. Fix bugs.
   3. Address user feedback.

The primary deliverables will be developed during 3.a, though they will not be considered production level until after the closed beta. An overview of the deliverables is given below.

* **Text uploader** - Takes an input text and inserts it into an offline text library.
* **Text analyzer** - Uses natural language processing to extract the most meaning words from the input.
* **Top words view**: Views the top words in rank order.
* **Word frequency view**: Views a subset of the top words and their frequency throughout the text.
* **Sentiment view**: Displays the overall sentiment of the text.
* **User documentation**:Documentation in application that gives instructions to the user.

The following sections will details project timeline, project evaluation, and milestones.

## Evaluation Plan

The evaluation plan can be broken into its systems and goal components. A system level evaluation will focus on the data and the application, ensuring that the program functions. The goal level evaluation will evaluate if the application aids in domain specific language retention. The following sections will address these components in more detail.

### System

The project testing will consist primarily of inputting data with already known expected results and comparing it to the program output data. This can be done autonomously by generating from a pre-generated word list and using it as input.

Useability testing will be important to understand how a customer interacts with the application. It is possible that customers may not understand how to interpret the results from the application, or what the application is capable of. User testing will intend to uncover any of these issues.

### Goal

The most direct method to test the benefit of the tool is through a scientific trial. A test group will be given a text to read in a target language and access to the learning tool whereas a control the control group will be given the text without the tool. Both groups will be required to read the text in the allotted time and then given a comprehension test after. If the hypothesis holds true, the test group will display a greater degree of comprehension.

## Resources and Costs

### Programming Environment

### Environment Costs

### Human Resource Requirements

## Timeline and Milestones

# Section C - Implementation Report

## Data Methods

## Datasets

## Analytics

## Data Cleaning

## Real-Time Queries

## Adaptive Element

## Outcome Accuracy

## Security Measures

## Product Health Monitoring

## Dashboard

# Section D - Post Implementation Report

## Project Purpose

## Datasets

## Data Product Code

## Hypothesis Verification

## Effective Visualizations and Reporting

## Accuracy Analysis

## Application Testing

## Application Files

## User’s Guide

## Summation of Learning Experience

# Section E- Sources

# Section F - Professional Communication