Natural Language Processing for Mental Health Risk Prediction

Requirements Standard Plan

Michael Breau, Jessica Dawson, Benjamin Chinnery, Yaruo Tian, Matthew Curtis

Version 1, 2024-03-06

Table of Contents

Academic Integrity Disclaimer

We would like to acknowledge that as a dedicated students of McMaster University, we have thoroughly read and comprehended the Academic Integrity Policy published by the university. We are committed to upholding the principles of academic honesty and integrity in all aspects of our educational journey. We understand the importance of acknowledging the work and ideas of others, and we pledge to ensure that all our academic endeavors are conducted with the utmost originality and compliance with the university's policy.

We affirm that the content presented in this document is entirely our own, and any external sources used have been appropriately cited and referenced.

Michael Breau

As I submit my work, I, **Michael Breau**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

Jessica Dawson

As I submit my work, I, **Jessica Dawson**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

Benjamin Chinnery

As I submit my work, I, **Benjamin Chinnery**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

Yaruo Tian

As I submit my work, I, **Yaruo Tian**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

Matthew Curtis

As I submit my work, I, **Matthew Curtis**, take full responsibility for the integrity of my work and promise to avoid any form of plagiarism, cheating, or dishonest behavior. This acknowledgment serves as a testament to my dedication to academic excellence and the fostering of a trustworthy academic community at McMaster University.

Control Information

Version	Delivery		Feedback	
	Deadline	*Delivered*	*Received*	*Integrated*
V1	October 6, 2023	October 6, 2023		
V2				
V3				

(G) Goals

Control Information

Table 1. Natural Language Processing for Mental Health Risk Prediction — Versionning Information — Goal Book

Section	Version	Lead	Delivered	Reviewer	Approved
G.1	1.0	Matthew Curtis	2023-10-06	Jessica Dawson	October 6, 2023
G.2	1.0	Matthew Curtis	2023-10-06	Jessica Dawson	October 6, 2023
G.3	1.0	Michael Breau	October 2, 2023	Jessica Dawson	October 6, 2023
G.4	1.0	Michael Breau	October 2, 2023	Jessica Dawson	October 6, 2023
G.5	1.0	Jessica Dawson	October 6, 2023	Michael Breau	October 6, 2023
G.6	1.0	Jessica Dawson	October 6, 2023	Michael Breau	October 6, 2023
G.7	1.0	Matthew Curtis, Michael Breau	2023-10-06	Jessica Dawson	October 6, 2023
G.4	1.1	Michael Breau	November 2, 2023	Jessica Dawson	November 2, 2023

(G.1) Context and Overall Objectives

As society advances so does the integration of technology into our lives. Many people struggle with mental health issues that go unnoticed, and undiagnosed as well, and are isolated so they never get the diagnosis and the support they need. In response to those two points working in tandem, Dr.Mosser's past teams, along with support from UQAM and several psychologists from around the world, have developed methods and approaches to assist psychiatrists with diagnosing mental health issues. Our goal for this project is to enter a competition hosted by eRisk where we will develop a Natural Language Processing (NLP) architecture that will be used to aid health care providers in early risk detection of certain mental health issues and conditions from data collected over the internet. The internet and social media posting can be the cause of many negative mental health symptoms, but, with the amount of data available, we think we can use it to our advantage to help diagnose and detect these issues. We will be basing our NLP system off of the current works and methods developed over the past years by eRisk. This will be done in collaboration with Professor Mosser and individuals from the University of Montreal. There are a wide variety of use cases we may be working on, but, as of right now, the main information we know is that we will be given a mental health issue or situation where we will be required to make a NLP system in order to take data we are given and determine the risk of said mental health issue for the given user. Our goal is to help eRisk expand the number of use cases that their NLP systems are applicable for and cover a wider range of scenarios and personal profiles. They have a strong base of NLP systems and data to work with, but the greater the breadth of the research the more people it can help. Early intervention is very important for mental health issues, and for issues such as suicidal tendencies and online predatory tendencies, it can be a matter of life and death. That is why this work is so important, and we will be working on expanding the number of situations in which it will be able to help people.

(G.2) Current situation

Currently, it can be a very difficult task to diagnose people with mental health conditions. If you are suffering from a mental health condition, it can often be hard to even originally reach out for help, and it is easy to feel isolated and alone. Due to this, many people never reach out to anyone and suffer alone especially in this post pandemic world where people still often feel very isolated as a hangover effect of the pandemic isolation. That is why it is so important to be able to identify the people who need help and offer them said help. Currently there is no good system that will help you identify potential early warning signs of mental health issues based on a person's online posting history. eRisk has been working on developing an interdisciplinary research field that aims to be applicable to a wide range of situations and to many different personal profiles. These can be used to help identify potential online predators, people with suicidal inclinations, or people that are susceptible to depression. They are building their reach through research and NLP, but it is still in the early stages and is limited. What we will do is tackle a new mental health problem that they have yet to consider and build on their previous practices and improve it making it relevant for our mental health issue. The goal is all about expanding the use cases that this methodology and research is applicable for and that is what we will be providing.

(G.3) Expected Benefits

The implementation of a natural language processing model that can detect early signs of mental health issues will have multiple benefits to its stakeholders. The system when completed will be used to participate in the 2024 eRisk competition. eRisk is an organization whos goal is to research early risk detection on the internet (in the case of this system and competition, the project focuses primarily on mental health). By participating in this competition, the system aims to benefit eRisk and its participants by providing any information that could possibly provide useful for furthering this research topic.

The system will be used to process data taken from an individuals messages (In the case of past competitions, the messages are normally taken from "reddit" users in communities that are related to the topic (Ex: r/ProblemGambling for detecting early signs of pathological gambling)). The system intends to benefit individuals who are affected by mental health issues by detecting early signs of these issues before they lead to a point that is irrecoverable for the individual.

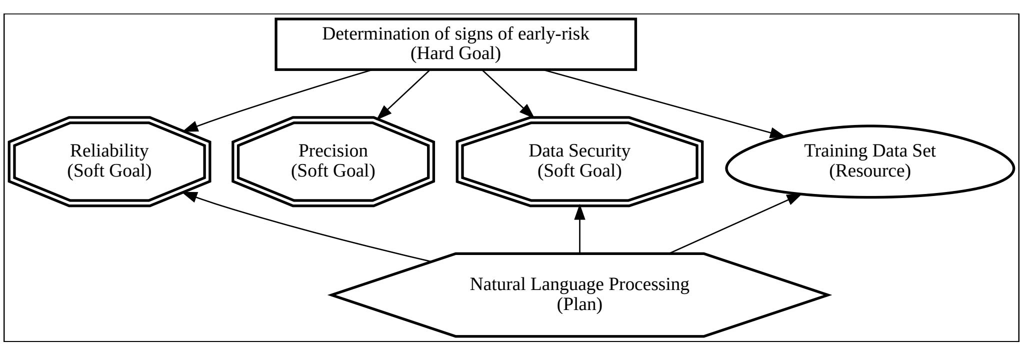

Figure 1. Goal Model Diagram for Natural Language Processing for Mental Health Risk Prediction

(G.4) Functionality overview

The requirements listed represent the core functionality of the Natural Language Processing system used to detect early signs of mental health issues. Detailed descriptions are available in the system section of this document.

Functional Requirements

- **Process Training Data:** The system will process training data to be used later on for natural language processing.
- **Natural Language Processing:** The system will process data using a natural language processing model to determine if the messages exhibit early signs of mental health issues.
- **Precise Detection of Early Signs**: The system should replicate similar key performance indicators to be competitive with past eRisk competition participants in various detection tasks (less than 100% difference).

Non-Functional Requirements

- **Security of Data**: Due to the nature of the data used with training as well as for detection purposes, the system should be free from security risks that could compromise the data.

(G.5) High-level usage scenarios

Use Case 1: Process Data

Goal:	To convert raw language data into a form usable by prediction models
Actor:	Developer
Precondition:	The developer has access to raw language data
Main Pattern:	• The developer submits the raw data to the system • The system returns a processed form of the data usable by prediction models to the developer

Use Case 2: Train Model

Goal:	To train a prediction model using processed data
Actor:	Developer
Precondition:	The developer has access to processed data with expected outputs

Main Pattern:	• The developer submits the data and expected outputs to the system • The system returns a trained prediction model to the developer • The system returns accuracy measures for the model’s predictions on the training data

Use Case 3: Predict Risk

Goal:	To predict a person’s mental health risk from new data
Actor:	Developer
Precondition:	The developer has access to a trained prediction model and processed data the model has not seen yet
Main Pattern:	• The developer submits the trained model and data to the system • The system returns predictions of mental health risk for each data point

Use Case 4: Measure Accuracy

Goal:	To compute accuracy measures for predictions made on data
Actor:	Developer
Precondition:	The developer has access to predictions made by a model and the corresponding expected values
Main Pattern:	• The developer submits the predictions and expected values to the system • The system returns a set of accuracy measures for the model’s performance

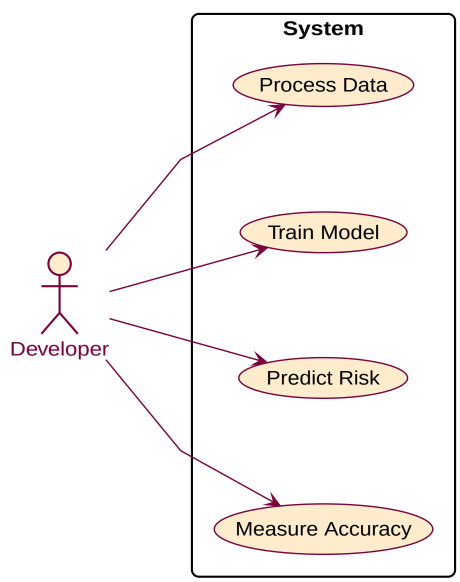

Figure 2. High Level use cases diagram

(G.6) Limitations and Exclusions

The system need not address the acquisition of raw language data, this functionality will be handled by a system developed by capstone team 8, who are working on the ops side of this project.

The system need not address providing mental health professionals access to the models. The project goal is to submit our findings to the eRisk competition, adapting the research into systems that mental health professionals can use is outside the scope of the project.

(G.7) Stakeholders and requirements sources

Table 2. Stakeholders

Stakeholder		Persona	Category
Direct Users	Psychiatrists	[Jessica]	Direct
	Police Officers	[Joe]	Direct
	Family Doctors	[Julia]	Direct
eRisk Competition Organizers		N/A	Direct
University of Montreal NLP Team		N/A	Direct
Mental Health Capstone Ops Team		N/A	Direct
Local Community			Indirect

Direct Stakeholders

Direct stakeholders have an interest in the implementation and can be used to elicit requirements. The groups that will be using the eRisk data and NLP architecture directly have been provided with personas to aid with requirement elicitation. They will be the primary users and will use the architecture in order to screen individuals to hopefully discover potential early signs of different mental health disorders. Although each group may use the NLP architecture to uncover different mental health issues and their motives will be different from one another.

Potential Stakeholders:

Jessica (Psychiatrist Persona):

Jessica is a 45-year-old Psychiatrist that works at her private practice. Her patient is a 14-year-old girl named Avery, and her mother has come to Jessica saying she is worried about Avery's mental health and thinks she may be beginning to suffer from depression and suicidal tendencies. When Avery comes into the doctor's office, she agrees to speak with Jessica about the topic but is very shy and reserved. Understandably so since the topic can be a hard and sensitive topic to approach. She ends up not really giving Jessica any information about how she is feeling and Jessica is left with no way of understanding if she is at risk or not. In moments like this Jessica loves to use eRisks NLP architecture to look at Avery social media accounts and determine if the model thinks she would be identified as someone that may be at risk for these issues. Without this model, Jessica is unsure of other ways to get more information about Avery's situation, but the NLP model can provide some useful insight for Jessica. As a result, she can bring up the topic of Avery's social media to her and create an open discussion about the feelings she may be having.

Joe (Police Officer Persona):

Joe is a 40-year-old police officer working in Hamilton, Ontario. He is on duty one night and gets a call from a distressed teenager. The teenage girl says that her friend's father has been acting very strange around her and is being creepy. This is all the information Joe is provided with and is stuck on what his next steps should be. Thankfully, he has the NLP architecture model from eRisk as a tool he uses when he gets calls like these or others about potential signs of mental health problems people are exhibiting. This is a key tool to see potential signs of early risk of mental health issues before drastic actions are taken and also helps find potential dangerous situations due to mental health issues exhibited online. When he runs the NLP model for this father, it comes back with a high risk of predatory behavior for him. This helps solidify to Joe that this is not just a one-off case and needs to be investigated swiftly before anything bad happens.

Julia (Family Doctor Persona):

Julia is a 35-year-old family doctor at her local doctors office. She knows the importance of mental health and how it can be hard for people to come forward and seek the help they need whether it is due to being shy, embarrassed, or any other number of reasons. As a result, she runs the eRisk NLP model on her patients the day before they come in. By doing this, she can find if there are any potential early signs of mental health issues and, if so, she can propose starting an open dialogue with her patient when they meet or ask her patient if they have been experiencing any feelings recently that would be related to said mental health issue. A lot of the time people just need an opportunity to speak about what they are going through, and this is a good way to get the dialogue started.

eRisk Competition Organizers:

The eRisk organizers will compare our systems performance in a variety of metrics in a competition format against other teams in order to further the research in the early risk prediction on the internet field.

University of Montreal NLP Team:

As previous competitors in the competion as well as mentors to the current project, the UoM NLP team has a direct interest in the implementation of the project.

Mental Health Capstone supervisors to the current project:

The projects supervisor's have a direct interest in the implementation as well as the outcome at the competition from the project.

Indirect Stakeholders

Local Community:

In the event that the project leads to the direct users, a change of an environment due to sucessful detection of early risk signs could provide a benefit to the local community.

(E) Environment

Control Information

Table 3. Natural Language Processing for Mental Health Risk Prediction — Versionning Information — Environment Book

Section	Version	Lead	Delivered	Reviewer	Approved
E.1	1.0	Benjamin Chinnery	October 2, 2023	Michael Breau	October 2, 2023
E.2	1.0	Yaruo Tian	October 2, 2023	Michael Breau	October 2, 2023
E.3	1.0	Yaruo Tian	October 2, 2023	Michael Breau	October 2, 2023
E.4	1.0	Yaruo Tian	October 2, 2023	Michael Breau	October 2, 2023
E.5	1.0	Benjamin Chinnery	October 2, 2023	Michael Breau	October 2, 2023
E.6	1.0	Benjamin Chinnery	October 2, 2023	Michael Breau	October 2, 2023

(E.1) Glossary

- **Natural Language Processing (NLP)**: A subfield of computer science concerned with giving computers the ability to understand and manipulate human language.
- **Neural Networks**: Sometimes referred to as Artificial Neural Networks, are machine learning systems modeled on the neural pathways of the human brain. This type of system endeavors to recognize underlying relationships in a set of data.
- **Mental Health**: Mental health encompasses emotional, psychological, and social well-being, influencing cognition, perception, and behavior
- **Training Models**: Data provided to machine learning systems to help teach it correct assumptions about a given topic
- **Early Risk Detection Error (ERDE)**: An Evaluation Metric that evaluates the number of writings processed by the algorithm before producing a positive prediction
- **eRisk**: eRisk organizes a competition every year to compete various Artificial Intelligences against each other to help identify mental health issues. We will be entering our NLP into their contest this year, and as a result are tuning our algorithm to their specifications.

(E.2) Components

- **ERisk Servers**: The system must utilize servers provided by ERisk during the testing stage to interatively get user writings and send responses.
- **Stanford CoreNLP Client**: The system could interact with Stanza's Java toolkit, Stanford CoreNLP, with its server interface. Doing so will allow the system to derive linguistic annotations for text,

tokenization and parsing in python.

- **Text Data and Profiles**: The data sources and profile information mainly come from various platforms such as Reddit. Using their APIs will support in NLP development.
- **Data Visualization Tools** Incorporating data visualization and analysis APIs and libraries such as Matplotlib and Pandas will aid in testing phase to generate an more accurate result.

(E.3) Constraints

- **Resource Constraint**: The final NLP models must be able to run on Canada Compute Servers without external processes running in conjunction.
- **Data Regulatory Constraint**: The system must be built to handle and process different sizes of data provided by ERisk and process them while abide by the set of rules that eRisk defined, such as the General Data Protection Regulation (GDPR).
- **Time Constraint**: The system must be built following the schedule of the ERisk's important dates, such as releasing training data and deployment on servers.
- **Compatibility** The system must follow ERisk's specification during deployment and testing phases in order to qualify for the program.

(E.4) Assumptions

- **Availability of ERisk Servers**: It is assumed that ERisk testing and deployment servers are active and available at all times during the testing and deployment phase.
- **Accuracy of User Profiles**: The assumption is made that all profile data provided for testing are formatted correctly and will be able to be processed without any errors, disruptions or further development procedures beyond the scope.
- **Disruption of ERisk's Schedule**: It is assumed that ERisk's schedule, including the date and time provided for important dates, are executed on time and in sync with the development process.
- **Connection Stability**: It is assumed that the connection between the system and ERisk labs' servers is smooth and stable, without connectivity disruptions from internet service providers.
- **Consistent Development Platforms**: The assumption is made that documentation and utilization will be consistent throughout the development process without sudden changes in stakeholders or business plans.

(E.5) Effects

- **Workload of Mental Health Specialists**: The system could help ease up on the workload of Mental Health Professionals by being an additional tool they could use which could help provide support with diagnosis.
- **Barriers to Accessing Mental Health Services**: The online and virtual nature of the system was

designed to help ease the barriers to support that people in remote communities often face.

- **Improved Detection**: The system may result in better detection of symptoms at varying stages by altering mental health professionals of signs that they might have missed.
- **Improved Treatment Plans**: Improved detection may result in mental health professionals being able to create more efficient treatment plans.
- **Cost of Mental Health Services**: The assistance and accessibility of these tools may reduce the overall cost of certain Mental Health Diagnostic Services.

(E.6) Invariants

- **User Data Privacy**: The system must maintain the privacy and anonymity of the individuals involved in either the training set or in any writings it evaluates.
- **Inputted Data Privacy**: The system must maintain privacy for any evaluations done for a user, by only sharing that information to the appropriate mental health assessor of the user.
- **Authority of Mental Health Professionals**: The system must maintain the current status quo for Mental Health Professionals being the only ones qualified to give a definitive diagnosis. This system is meant to aid the process, it is not meant to be the sole authority.
- **Ethical Guideline Accountability**: The system must act in accordance with the proper ethical standards and practises provided to us from the correspondent mental health associates.
- **Data Retention Guidelines**: The system must act in accordance with the nesscessary data retention guidelines of the appropriate environment, only saving and storing what is allowed.

(S) System

Control Information

Table 4. Natural Language Processing for Mental Health Risk Prediction — Versionning Information — System Book

Section	Version	Lead	Delivered	Reviewer	Approved
S.1	1.0	Michael Breau	October 5, 2023	Jessica Dawson	October 6, 2023
S.2	1.0	Michael Breau	October 5, 2023	Jessica Dawson	October 6, 2023
S.3	1.0	Jessica Dawson	October 6, 2023	Michael Breau	October 6, 2023
S.4	1.0	Jessica Dawson	October 6, 2023	Michael Breau	October 6, 2023
S.5	1.0	Michael Breau	October 5, 2023	Jessica Dawson	October 6, 2023
S.6	1.0	Jessica Dawson	October 6, 2023	Michael Breau	October 6, 2023
S.2	1.1	Michael Breau	November 2, 2023	Jessica Dawson	November 2, 2023
S.2	1.2	Jessica Dawson	March 6, 2024	Michael Breau	March 6, 2024

(S.1) Components

- **Text Pre-Processing Component**: Text is processed with the following methods: Tokenization, Lowercasing, Stopword Removal, and Stemming or Lemmatization.
- **Vectorization Component**: Converting words/tokens into numerical vectors.
- **Prediction Component**: Generate a prediction on wether there are early signs of gambling addiction present in the text.
- **Output Component**: Due to a lack of a need for usability, the output will be printed to the console.

(S.2) Functionality

This section describes in further detail the components listed in the previous section by describing their functional and non-functional requirements.

General and Security

Non-Functional Requirements:

- GR1: The system should be compatible on MacOS, Windows, and Linux.
- GR2: Copyrighted content should not be included in the output without reference.
- SR1: The system will be tested periodically to detect a potential lack of processing power.

- SR2: Sensitive user data must not be present within the results generated.

Each task will have their own accuracy metrics that the quality of the models predictions will be evaluated on.

Task 1: Search for Symptoms of Depression Component

Functional Requirements:

- T1FR-1: The system will be able to parse both xml and jsonl files.
- T1FR-2: The system will rank input sentences from users based on their correlation to 21 depression symptoms from the beck depression index.
- T1FR-3: The system will send the predictions to a txt file in the format "{symptom_number}, Q0, {sentence-id}, {position_in_ranking}, {score}, {system_name}".

Task 2: Early Detection of Signs of Anorexia Component

Functional Requirements:

- T2FR-1: The system will be able to parse both xml and jsonl files.
- T2FR-2: The system will make predictions given a set of a user’s posts.
- T2FR-3: The system will allow posts to be added to a user’s set of posts and the system will update the prediction for that user.
- T2FR-4: The system will send the predictions to a txt file in the format "{username} {prediction (0 or 1)}" with each entry being on a separate line.

Task 3: Measuring the Severity of the Signs of Eating Disorders Component

General Functional Requirements:

- T3FR-1: The system will be able to parse jsonl files.
- T3FR-2: The system will make predictions given a set of a user’s posts.
- T3FR-3: The system will send the predictions to a txt file in the format "{username} {prediction (0-6) for Q1} … {prediction (0-6) for Q28}" with each entry being on a separate line.

Accuracy Functional Requirements:

These functional requirements cover how the accuracy of the model will be evaluated. Some background information on chosen accuracy metrics and evaluation methodology will be supplied.

The following is taken from [3] and discusses the metrics that will be used to evaluate acuracy for task 3:

Evaluation is based on the following effectiveness metrics:

- Mean Zero-One Error (*MZOE*) between the questionnaire filled by the real user and the questionnaire filled by the system (i.e. fraction of incorrect predictions).

$$MZOE(f,Q) = \frac{|\{q_i \in Q : R(q_i) \neq f(q_i)\}|}{|Q|}$$

  where f denotes the classification done by an automatic system, Q is the set of questions of each questionnaire, q_i is the i-th question, $R(q_i)$ is the real user's answer for the i-th question and $f(q_i)$ is the predicted answer of the system for the i-th question. Each user produces a single *MZOE* score and the reported *MZOE* is the average over all *MZOE* values (mean *MZOE* over all users).

- Mean Absolute Error (*MAE*) between the questionnaire filled by the real user and the questionnaire filled by the system (i.e. average deviation of the predicted response from the true response).

$$MAE(f,Q) = \frac{\Sigma_{q_i \in Q}|R(q_i) - f(q_i)|}{|Q|}$$

  Again, each user produces a single *MAE* score and the reported *MAE* is the average over all *MAE* values (mean *MAE* over all users).

The following measures are based on aggregated scores obtained from the questionnaires. Further details about the EDE-Q instruments can be found in appendix A.

- Restraint Subscale (RS): Given a questionnaire, its restraint score is obtained as the mean response to the first five questions. This measure computes the RMSE between the restraint ED score obtained from the questionnaire filled by the real user and the restraint ED score obtained from the questionnaire filled by the system. Each user u_i is associated with a real subscale ED score (referred to as $R_{RS}(u_i)$) and an estimated subscale ED score (referred to as $f_{RS}(u_i)$). This metric computes the RMSE between the real and an estimated subscale ED scores as follows:

$$RMSE(f,U) = \sqrt{\frac{\Sigma_{u_i \in U}(R_{RS}(u_i) - f_{RS}(u_i))^2}{|U|}}$$

  where U is the user set.

- Eating Concern Subscale (ECS): Given a questionnaire, its eating concern score is obtained as the mean response to the following questions (7, 9, 19, 21, 20). This metric computes the RMSE between the eating concern ED score obtained from the questionnaire filled by the real user and the eating concern ED score obtained from the questionnaire filled by the system.

$$RMSE(f,U) = \sqrt{\frac{\Sigma_{u_i \in U}(R_{ECS}(u_i) - f_{ECS}(u_i))^2}{|U|}}$$

- Shape Concern Subscale (SCS): Given a questionnaire, its shape concern score is obtained as the mean response to the following questions (6, 8, 23, 10, 26, 27, 28, 11). This metric computes the RMSE between the shape concern ED score obtained from the questionnaire filled by the real user and the shape concern ED score obtained from the questionnaire filled by the system.

$$RMSE(f,U) = \sqrt{\frac{\Sigma_{u_i \in U}(R_{SCS}(u_i) - f_{SCS}(u_i))^2}{|U|}}$$

- Weight Concern Subscale (WCS): Given a questionnaire, its weight concern score is obtained as the mean response to the following questions (22, 24, 8, 25, 12). This metric computes the RMSE between the weight concern ED score obtained from the questionnaire filled by the real user and the weight concern ED score obtained from the questionnaire filled by the system.

$$RMSE(f,U) = \sqrt{\frac{\Sigma_{u_i \in U}(R_{WCS}(u_i) - f_{WCS}(u_i))^2}{|U|}}$$

- Global ED (GED): To obtain an overall or 'global' score, the four subscales scores are summed and the resulting total divided by the number of subscales (i.e. four). This metric computes the RMSE between the real and an estimated global ED scores as follows:

$$RMSE(f,U) = \sqrt{\frac{\Sigma_{u_i \in U}(R_{GED}(u_i) - f_{GED}(u_i))^2}{|U|}}$$

The accuracy of the model will be evaluated by comparing between the value of these metrics for the model's predictions and the values of these metrics for answering zero for all questions, answering six for all questions, or answering the average score for each question. If the model fails to outperform any of these guessing strategies in any metrics then it is not providing any increase in accuracy.

- T3FR-4: The model will outperform guessing all zeros on at least one of the above accuracy metrics.
- T3FR-5: The model will outperform guessing all sixes on at least one of the above accuracy metrics.
- T3FR-6: The model will outperform guessing the average on at least one of the above accuracy metrics.

(S.3) Interfaces

Interface	Description
pre_process(raw_text_data, settings[]) → returns processed_data	Tokenizes, converts to lowercase, removes stopword tokens, and reduces tokens to root forms in order to convert raw text data into data that can be vectorized. Settings specify the processing that takes place.
vectorize(processed_data, settings[]) → returns model_data	Converts processed data into numerical vectors that a model can be trained on. Settings specify the processing that takes place.

train_model(model, model_data, settings) → returns model	Trains a given model on model_data, returns the model after training. Settings specify how the model is formed and trained.
predict(model, test_data) → returns predictions	Runs a trained model on test_data, returns the model's predictions.
measure(predictions, expected_values) → returns measures[]	Takes in predicted values and the corresponding expected values, returns a set of accuracy measures detailing the performance of the predictor.

(S.4) Detailed usage scenarios

Use Case 1: Process Data

1. A developer has some raw text data.
2. The developer calls pre_process() and passes it this data and a list of settings.
   a. If pre_process() fails to process the data the developer receives an error message describing what went wrong. The developer reformats the data and/or adjusts settings and then returns to step 2.
3. The developer receives data processed into a format specified by the given settings.

Use Case 2: Vectorize Data

1. A developer has some processed data.
2. The developer calls vectorize() and passes it this data and a list of settings.
   a. If vectorize() fails to process the data the developer receives an error message describing what went wrong. The developer reformats the data and/or adjusts settings and then returns to step 2.
3. The developer receives data vectorized into a format specified by the given settings.

Use Case 3: Train Model

1. A developer has some vectorized data.
2. The developer has an untrained model.
3. The developer calls train_model() and passes it the untrained model, the data, and a list of settings.
   a. If the model fails to train effectively on the data the developer receives an error message describing what went wrong. The developer returns to step 3 and adjusts settings accordingly.
   b. If the model can not train on the given data the developer receives an error message describing what is wrong with the data. The developer reformats the data and returns to step 3.
4. The developer receives a model trained based on the given settings.
5. The developer also receives some statistics on the model's training performance.

Use Case 4: Predict

1. A developer has a trained model.
2. The developer has some test data the trained model has not yet seen.
3. The developer calls predict() and passes it the model and the data.
    a. If the model can not use the given data the developer receives an error message describing what is wrong with the test data. The developer reformats the test data and returns to step 3.
4. The developer receives a set of predictions the model made.

Use Case 5: Measure

1. A developer has some predicted values.
2. The developer has the corresponding expected values.
3. The developer calls measure() and passes it the predicted and expected values.
    a. If the two sets of values do not align with eachother (mismatched lengths, format, etc.) the developer receives an error message describing what is wrong with the data. The developer reformats or re-acquires the data and returns to step 3.
    b. If the format of the data can not be processed into a particular measure the developer receives an error message describing what is wrong with the data. The developer reformats or re-acquires the data and returns to step 3.
4. The developer receives a set of accuracy measures detailing how close the predicted values are to the expected values.

(S.5) Prioritization

Due to the nature of the system, the different components are all dependent on eachother and therefore their requirements are all high priority with little room for dropping functionality in the pre-Processing component of the system.

Text Pre-Processing Component	must-have	should-have	could-have	won't-have
The system will divide text into individual units of words and or sub-words (Tokenization).	**x**			

The system will convert generated tokens into lowercase to preserve consistency.		**x**		
The system will remove stopword tokens (Common words that do not commonly have an effect on the meaning).		**x**		
The system will reduce tokens to their root forms (Ex: "moving" to "move").		**x**		
The system will produce an output that is useable by the Vectorization Component.	**x**			
Vectorization Component	must-have	should-have	could-have	won’t-have
The system will convert tokens into numerical vectors.	**x**			
The system will produce an output that is useable by the Prediction Component.	**x**			
Prediction Component	must-have	should-have	could-have	won’t-have

The system will make predictions using the vectorized tokens, the trained model, as well as with the created prediction algorithm.	**x**			
The system will produce an output that is useable by the Output Component.	**x**			
Output Component	must-have	should-have	could-have	won't-have
The system will produce the output of the predictions to the console.	**x**			
The system will produce output to the console that is understandable by the project team.	**x**			

(S.6) Verification and acceptance criteria

- Data processing functions (pre_process, vectorize) will be deemed satisfactory if their actual output given some test data aligns with the expected output based on the settings they are given. This will be determined through unit tests.
- The data processing pipeline will be deemed satisfactory if, when given some test data, the actual output of each step in the pipeline matches the expected output. This will be determined through a series of unit tests.
- The accuracy measure calculations will be deemed satisfactory if, when given some test data, the resulting accuracy measures match pre-calculated expected values. This will be done through unit tests.
- The model training functions (train_model) will be deemed satisfactory if the model managed to achieve an accuracy on the eRisk training data comparable to a control accuracy (ie. rate of successful diagnoses by mental health professionals, accuracy of a different, comparable model).
- The entire system will be deemed satisfactory if the resulting models manage to achieve an accuracy on the eRisk testing data comparable to a control accuracy (ie. rate of successful diagnoses by mental health professionals, accuracy of a different, comparable model).

(P) Project

Control Information

Table 5. Natural Language Processing for Mental Health Risk Prediction — Versionning Information — Project Book

Section	Version	Lead	Delivered	Reviewer	Approved
P.1	1.0	Matthew Curtis	2023-10-06	Jessica Dawson	October 6, 2023
P.2	1.0	Matthew Curtis	2023-10-06	Jessica Dawson	October 6, 2023
P.3	1.0	Yaruo Tian	October 5, 2023	Michael Breau	October 5, 2023
P.4	1.0	Yaruo Tian	October 5, 2023	Michael Breau	October 5, 2023
P.5	1.0	Yaruo Tian	October 5, 2023	Michael Breau	October 5, 2023
P.6	1.0	Benjamin Chinnery	October 4, 2023	Michael Breau	October 5, 2023
P.7	1.0	Benjamin Chinnery	October 4, 2023	Michael Breau	October 5, 2023

(P.1) Roles and personnel

- **Developers**: Matthew Curtis, Jessica Dawson, Michael Breau, Benjamin Chinnery, Yaruo Tian

We will be working with professors and many different other teams throughout our project with our project as one specific task (Developing a NLP architecture). We are unsure of the exact specifics for our projects as of right now due to the fact eRisk has not released the topics that we will be choosing from for the 2024 competition we will be taking part in. As a result, each of our main roles at the moment are just general developers.

- **GitHub Issue Tracker**: Michael Breau

Michael will be responsible for organizing and managing all of our github issues. This includes creating issues, creating the respective tags and categories and also making sure they are closed when they are finished.

- **Team Liaison**: Benjamin Chinnery

Benjamin will be our team liaison and as a result he will be the primary person reasonable for organizing our meeting as a group with all of our input put into consideration. He will also be our primary contact between our team and Professor Mosser along with the other teams involved as well. We all will of course have access to talk with Professor Mosser and the other teams and will do so. Benjamin will just act as our primary point of contact.

(P.2) Imposed technical choices

Our prior choices were largely decided through technical analysis of the tools, industry best practices, company policies and relevance, and familiarity with the tools.

- **Programming Language**: We will be primarily using Python for our main programming language. This is the most common language to be used for machine learning and all of the previous coding infrastructure we have to work with is written in Python as well so Python is an obvious and necessary choice for our programming language.
- **Version Control**: GitHub will be our version control vehicle of choice for this project. We have a repository with just this group for the document and reporting side of the project and we also have a shared repository with Professor Mosser, the Ops team and the team in Montreal as well for our actual code.
- **Testing**: Pytest will be used to validate code.
- **Linter**: We will be using Pylint as our linter of choice to ensure code quality
- **Framework**: The frameworks we will be utilizing for this project will stem from what was used by Professor Mossers team in past iterations of this competition. We will be utilizing the Torch machine learning framework
- **Libraries**: There are a number of libraries that have been used for previous years iterations of this competition by Professor Mossers team. We plan on sticking with these as best practices and to keep consistency. These Libraries include pandas for data manipulation and analysis, Gensim for a number of NLP functionalities and scikit-learn for machine learning as well.

(P.3) Schedule and milestones

- Task 1: Learn required knowledge for project development such as Natural Processing Techniques and its corresponding libraries.
- Task 2: Initial setup and install required tools.
- Task 3: Use the Pathological Gambling Project as reference for initial prototype.
- Task 4: Develop version 1 (functional prototype) for proof of concept.
- Task 5: Train NLP model with data provided by ERisk.
- Task 6: Analyze result and modify version 1 of product.
- Task 7: Develop design documentation for futher product revision.
- Task 8: Develop version 2 (final product) and test for accuracy.

(P.4) Tasks and deliverables

Iteration #1: Weeks 1-4

- Task 1: Learning the required Natural Language Processing techniques is essential for development. As the majority of the product is focused on the AI aspect with minimal front end component, all developers are required to know NLP techniques and how to implement them.
- Task 2: Installing the required tools and setting up the environment will streamline NLP projects as they require strict reproducibility, performance optimization and dependency management.

Iteration #2: Weeks 5-8

- Task 3: Using the Pathological Gambling Project as reference for initial prototype will not only help developers familiarize with the NLP procedures but also further mimic and improve the exisiting project.
- Task 4: Develop the initial functional aspect of the system to process basic data. This task should be completed prior to the proof of concept.

Iteration #3: Weeks 9-12

- Task 5: Further develop the product until it could train using the data provided by ERisk. The result should yield 50% of the expected outcome with moderate accuracy.
- Task 6: Taken the results into consideration, modify the system according to the results from previous task. The modified system should achieve a result greater than 50% of the expected outcome.

Iteration #4: Weeks 13-16

- Task 7: Develop design documentation to further analyze aspects of improvement for the system.
- Task 8: Modify and test the system to yield data with a high degree of accuracy. The system should be able to process training data efficiently and yield 90% of the expect outcome.

(P.5) Required technology elements

The successful development and deployment of the Natural Lanugage Processing for Pathological Gambling project will require the following software/hardware tools and specifications:

1. Most updated version of Python
2. Python Libraries: pandas, Gensim, and scikit-learn
3. ERisk servers

(P.6) Risk and mitigation analysis

NLP Learning Curve

When the original description for the project was released to our group, it was clearly stated that this project would not require any prior experience with AI work, and would instead be a gradual learning process with a ramp up period. To help mitigate the risk of team members falling behind schedule, regular progress status check-ins in our team meetings could help mitigate difficulty. Solutions to help out teammates could come from other members of the group sharing their understanding, or which resources they used if they ran into similar problems. If team member support isn't sufficient, it is recommended for team members to reach out to project facilitators such as Dr. Mosser if further clarification is required.

Difficulty regarding eRisk Specification

Due to the ambiguity of the project specification in the early states, team members may feel overwhelmed with attempting to stay ahead of the curve in terms of preparing elements of the project before details regarding this year's competition are revealed. Team members are encouraged to keep up to date on all released eRisk specifications and familiarize oneself with the provided past documentation and specification from the prior teams. Additionally, risk mitigation can be attained by team members practising the exercises provided by Dr. Mosser and Dr. Meurs.

Difficulty Balancing Project Deliverables and Competition Deliverables

Due to this project having two independent timelines, one for the course deliverables and one for the contest deliverables, team members may experience difficulty balancing both responsibilities. The best way to manage this is to stick to clearly defined schedules and have fair divisions of work in order to ensure that all important project aspects are assigned to someone at all times.

(P.7) Requirements process and report

-**eRisk Competition Organizers**: Arrange internal team meetings and discussions to stay informed and up to date regarding any specifications and important deadlines for the eRisk Competition, in order to keep our project on track and to the right specifications. The competition organizers will release details such as what mental health issues will be focused on, timing and processing constraints, and general milestone deadlines. This information will be found on the website as well as communicated through their corresponding e-mail services.

-**University of Montreal NLP Team**: The team of researchers and experts working at the University of Montreal offer invaluable insight towards creating the project, as well as being major stakeholders for our team, as they have a major say in quality assessment of our project. Our team will have regularly scheduled meetings with the core Natural Language Processing Team as well as with any other collaborative experts that can provide feedback and instruction for us.

-**Mental Health Capstone Ops Team**: As this project is not solely the responsibility of 1 Capstone team, it is important that we as the Natural Language Processing team work closely with the Software Architecture Operations team to ensure team synergy, which is vital due to the direct reliance that both teams have with each other. Our primary means of accomplishing this is through regular inter-team communications and meetings to discuss any challenges or highlight key aspects of implementation to further each others understanding, as well as make sure that both projects are on the right track.

-**Mental Health Professionals**: Due to the scope of this project being so closely intertwined with the work of mental health professionals, it is imperative that we arrange to receive insight from people in that field. The overall prior team associated with this project is a widely multidisciplinary effort, and luckily for us, that team includes some therapists and other mental health workers from around the world who can provide us insight and feedback on our progress. It is important to put a strong emphasis on their outlook, as they are most closely related to the codes and standards that this project should follow in order to not violate any ethical standards.

-**Users of Online Mental Health Services**: It is important to conduct some user interviews with people who have either used virtual mental health services in the past or are interested in using them. Getting user feedback and input can further our understanding to give us a better idea of the virtual mental health landscape.

References

- [1] Bertrand Meyer. *Handbook of Requirements and Business Analysis*. Springer. 2022.
- [2] Ian Sommerville and Peter Sawyer. *Requirements Engineering: A good Practice Guide*. Wiley. 1997.
- [3] Javier Parapar, Patricia Martín-Rodilla, David E. Losada, and Fabio Crestan. *Overview of eRisk at CLEF 2022: Early Risk Prediction on the Internet (Extended Overview)*. Springer. 2022.