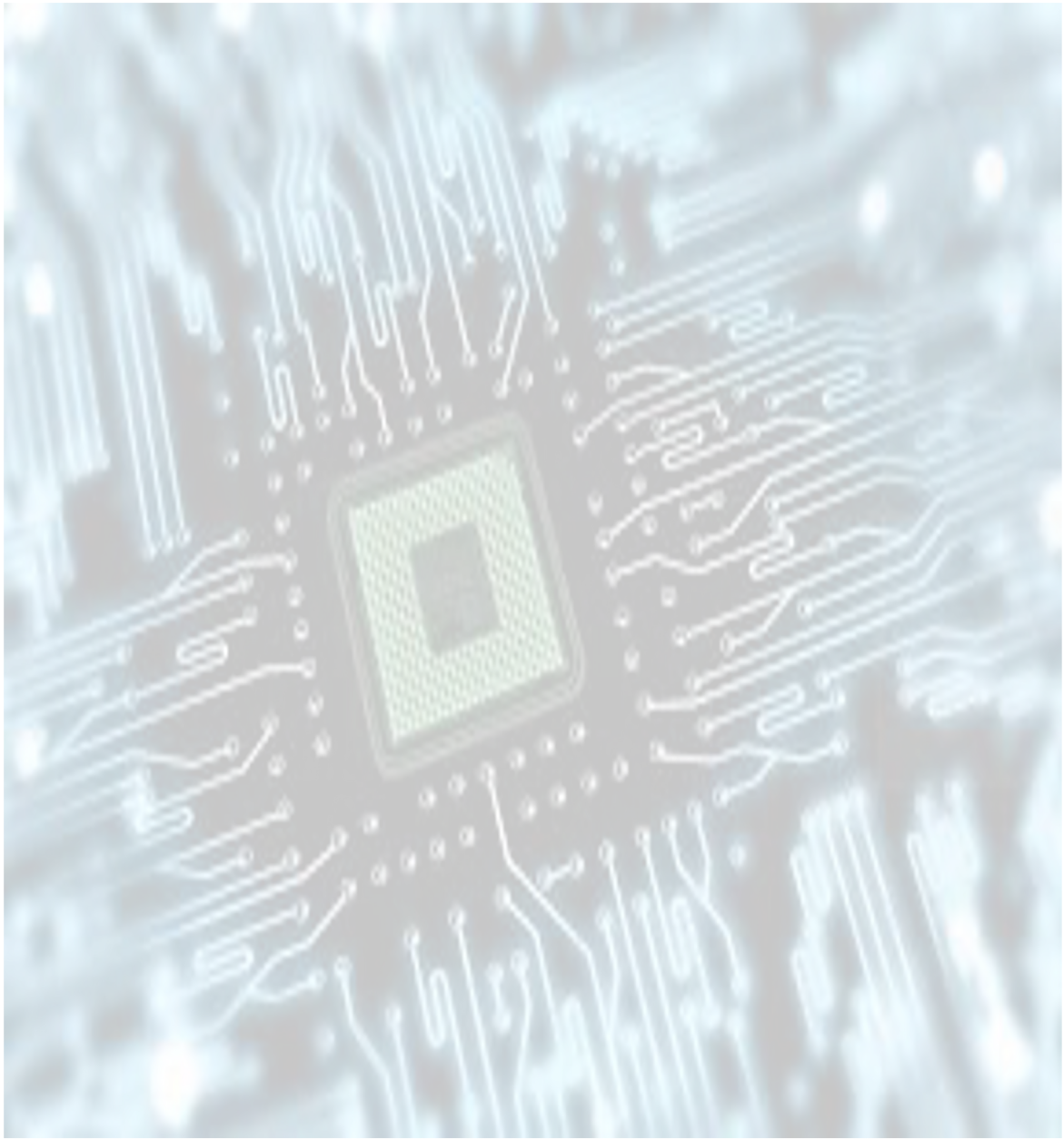
Smart Byte

# AIDI 2005-02: Capstone Warmup

# Next word predictor



January 31st 2023

Team members:


Raniya Alam || 100889828 || *Tech Lead / QA Lead*
Mehmet Burga Gunaydin || 100854077 || *Data base Lead*
Shenglin Qian || 100849798 || *Front End Developer*
Tyler Wilson || 100773241 || *Back End Developer*

# A next word predictor is a robust use of AI which can lead to direct user and business applications with the ability to extend to other languages and features

Smart Byte

## Business problem overview

- Aides to support writing are getting increasingly popular e.g. next word prediction and grammar support
- Next word prediction can help save time e.g. emails, texting etc.
- A next word predictor web app with easy integration can help users streamline their overall browsing and messaging experience
- Having a cloud based web app will also increase the scalability to additional features later on
- An enhanced next word predictor could even incorporate other up and coming technologies like Chat GPT or even additional languages. This could lead to business propositions e.g. marketing material creation/translation etc.

## Potential use cases

The next word predictor can be enhanced to handle various scenarios:

- Users feel supported with helpful recommendations when writing sentences. Which is a great timesaver and language aid
- Having a web app means that users will be able to easily bolt on to existing browsers without the need to pre-install software
- The basic functionality will be next word prediction in English. We will also evaluate extending functionality based on feasibility. :
  - **Option 1:** Providing another language e.g. Canadian French
  - **Option 2:** Providing additional libraries e.g. Shakespearean English
  - **Option 3:** Integrating with certain aspects of Chat GPT functionality



Note: This capstone topic is new for this group

# Our proposal is to design a cloud based web app with next word prediction capabilities based on user input in English as well as an additional model if feasible

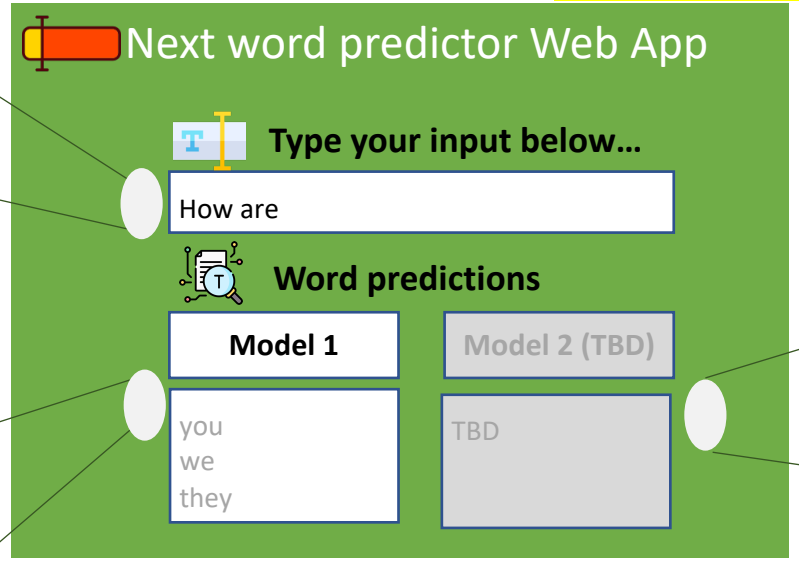
Smart Byte

**High level user journey:** A user will be able to open a cloud based web app on their browser and input English words. The web app will display predicted words back. If feasible an additional model will be included to demonstrate scalability of the product. The overall user experience will be defined which might include the ability to toggle between models depending on the option chosen. The practical application of the product will be to support users in writing materials more efficiently e.g. emails, essays etc.

**Note:** The scope might evolve with time, and any changes agreed with Product Manager. The key focus will be building a next word predictor for the English language. The additional model scope will be assessed as the project progresses and might be a proof of concept

- Users to provide initial input consisting of a few words in a dedicated web app hosted on a cloud server
- For model 1 this will be in English
- For model 2 this will be determined depending on feasibility and the option chosen

ILLUSTRATIVE




- The predicted words will appear after the user has entered the input
- We will also evaluate design options as part of the discovery phase to have a positive user experience e.g. having predicted words appear next to input words etc.

- Users will have the ability to toggle between the chosen model (TBD)
- Various options for a different language, writing style, integration with other AI tools are being considered

FOR DISCUSSION

# We have some initial thoughts on the key elements and the supporting technology stack but further guidance and research is required

Smart Byte

| | High level features | Tech options | Research questions (not exhaustive) |
|---|---|---|---|
| Web/E-commerce Application | • A stand along web app where users can input text and view predicted words | • Design a basic web application – easier integration<br>• Languages: HTML, PHP, Java script…… | • What options exist to build a simple custom web application for ease of integration and seamless user experience?<br>• What are the key use cases we would like to focus on?<br>• What are the basic features that need to built for Capstone 2? |
| Predictive model(s) | • Predictive language capabilities based on user input with reasonable level of accuracy | • TBD: Explore options around predictive/NLP modelling<br>• Languages: Java Script, Python.. | • What options exist to build and integrate a basic word prediction model?<br>• What language(s) should be used to build/customize the model?<br>• What are the additional models we could build (feasibility assessment) |
| Integration | • Able to integrate web app code (hosted on the cloud) with language model to have a seamless user experience | • Create code to integrate web app with language data models and predictive capabilities<br>• Languages/tools: Python, PHP, Java Script, Flask, Docker… | • Based on the options identified for the predictive model, supporting logic/features, and any additional language models how do we integrate all the elements, what coding language, functionality etc. is required? |
| Data set | • Explore use of existing NLP language libraries/available data sets to support predictive word capabilities<br>• Feasibility of additional model to be assessed | • Explore availability of language libraries<br>• Create selected prompts (TBD)<br>• Languages: Java Script, Python.. | • What are the minimum data requirements to train the prediction model and how do we determine and improve accuracy?<br>• What options are viable for the data set(s) – leverage open source, create a basic data set? |
| Hosting solution | • Cloud based solution which is publicly accessible<br>• Ability to link with GitHub repository | • Cloud hosting solutions e.g. Azure<br>• Ability to link with GitHub repository | • Across all the key elements and options what are the key consideration for the hosting solution<br>• Which hosting solution is the easiest/cost effective to store the application on? |



**Next steps:** Conduct a discovery phase to evaluate available options, refine scope and delivery plan

# The project plan will iteratively build the web application with regular reviews before launch in April. Scope to be revised if needed

INITIAL HIGH LEVEL PLAN

Smart Byte

| Stages | w.c. Jan 30 | w.c. Feb 6 | w.c. Feb 13 | w.c. Feb 20 | w.c. Feb 27 | w.c. Mar 6 | w.c. Mar 13 | w.c. Mar 20 | w.c. Mar 27 | w.c. Apr 3 | w.c. Apr 10 | w.c. Apr 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Discovery phase** | Finalize options for tech stack<br>Source/create data for English model<br>Initial solution design<br>◆ Jan 31st: Warm Up | | | | | | | | | | | |
| **MVP build** | Initial set up of web app<br>Build and train English prediction model<br>Identification of database platform and training data<br>Feasibility assessment of model 2 and solution design (TBD)<br>▲ Feb 14th: Project update meeting<br>◆ Feb 28th: MVP (1 week before or after is acceptable) | | | | | | | | | | | |
| **MMP build** | Next word web app and predictive model(s) tested, refined and finalized<br>Web app integrated on Azure (TBD) and available on team GitHub<br>▲ Mar 14th: Project update meeting<br>◆ Mar 28th: MMP | | | | | | | | | | | |
| **Final presentation and IT Expo** | IT Expo / final presentation materials prepared<br>Final presentation preparation along with demo and uploaded on GitHub<br>◆ Apr 18th: Final presentation<br>◆ Apr 11th: IT Expo | | | | | | | | | | | |

Key
▲ Update meeting with Marcos if needed
◆ Milestone