Golden Path Application Framework

The importance of abstracting the individual pieces of this solution is not just for better software design, but also to facilitate collaborative efforts between multiple people and or organizations. The basic premise of the Testing Framework is to provide an apples to apples comparison between AI translation solutions. Therefore the interface itself may be a more useful invention itself then the individual AI works as it provides a foundation for different ideas to be compared beyond speculation.

Please reference the enumerated block diagram in block_diagram.pdf.

1) **Translation Editor or AI Testing Framework Components:** This top level abstraction associates modules with the Translation Editor, or AI Testing Framework. The top level module division between the Editor/Testor and the AI Solution. This allows the Editor to make use of different possible AI Solutions and the Testing Framework to evaluate the performance of the same.

2) **Translation Editor or AI Testing Framework:** This block represents both software environments that a Bible Translator uses for translating scripture and the AI Testing Framework which evaluates and scores AI Solutions. The divisions here abstract the Editors/Tester from both the Target Language and the Golden Path Next Verse Suggester implementation. Looking outward from this block represents the API of the AI translation system. Likely implemented by a RESTful API. The methods associated with this module are

- **Save verse:** This publishes a translated verse by reference[1] into a separate database accessible by the AI solution.
- **Get AI Verse X:** This retrieves the AI suggestion for a specific verse translation.
- **Get next:** This requests the Golden Path module to suggest what verse should be translated next for optimum AI performance.

3) **Target Language Components:** This block abstracts the specifics to the Target Language from the Editor/Tester. The following two components are included.

4) **Published[2] Translated Scripture:** This is a publication of scripture passages by the Editors block does not replace the Editors various internal scripture representations, but is a publication of this data by the Translation Editor for the AI to use. This published work, may be in a different process and/or in the cloud with the rest of the AI Solution.

5) **Target Data abstraction/Tokenizer:** Different languages sometimes require different Tokenizers for breaking the language string into units such as words or lexicographical units. This module also

---

1 Each method which references a verse of scripture does so by a canonicalized reference string instead of broken out arguments of book chapter and verse. This allows future expansion for referencing segmented extra biblical information. Implementations are expected to be tolerant of unknown data references.

2 The word **published** is referencing the act of making the translated text available to the AI Solution, not the act of making the text available to the target community.

represents other processing steps which are language specific. This module is responsible for both Tokenizing the published translated scripture for the AI, and Detokenizing the suggested translation from the AI for the Editor or Testing Framework.

6) **Golden Path Next Verse Suggester:** This component is responsible for suggesting which verse should be translated next for maximum AI usefulness. This component is abstracted from the AI solution to allow the Testing Framework to test different Next Verse Suggesters against different AI Frameworks as well ensuring that the translator is never forced to follow the next verse suggestions despite which AI solution is employed.

7) **AI Solution Components:** This collection of modules represents a specific AI Solution. The data sources for the AI solution [9] other then the published Translated target language [4] are included within the abstraction allowing freedom in AI implementation as to what and how it uses existing data. Constraints are placed in this for the case of apples to apples comparison will be mentioned later.

8) **AI Solution X:** This component represents the various AI solutions which the [2] Testing Framework will evaluate. The AI Solution is responsible for generating tokenized Bible Verses for the Target Language as requested by Bible reference. The AI Solution can request previously translated scripture by reference from [4] the Published Translation which is delivered in tokenized form. Embeddings if used are hidden within the AI Solution and not part of its interface.

9) **Existing Translation Reference Data:** This abstraction provides data to the AI solution from existing resources such as existing translations of the Bible. If the reference system is extended to extra biblical texts, then this can also extend to other materials as well. For the sake of comparing the relative strength of different AI solutions and also excelerating the work of building new AI solutions, a common set of translations will be compiled. This set of data is abstracted separately from the AI solution so that the AI solution can be tested on different data-sets which may not be publicly available.

10) **Data abstraction/Tokenizer:** Similar to [5] the published data Tokenizer, this data abstractor provides a common API to the AI solution for accessing the data so that the AI solution is dataformat agnostic. It also tokenizes the data so that the AI solution is language agnostic. There is a different data abstraction written for each language and data format source.

11) **Existing Translation Texts:** These are the actual files of existing Translations. As mentioned above, a standardized set of these are to be collected for evaluating translations in and apples to apples context, but they are abstracted away from the AI solution so that the two solutions are not tied together. The translation text can be in any format which can be readily parsed and the requested verse obtained when requested by reference. The format variation is covered by the corresponding data abstraction.