Lomb

Structure

- What is our domain?
- What are some of the debates and previous work?
- Open research questions
- Implementation details

Understanding the domain

Stable Dependencies principle

"The dependencies between software packages should be in the direction of the stability of the packages. That is, a given package should depend only on more stable packages."

“Knowledge crunching”

The *domain* is the shared language between the experts in some area of knowledge or human activity and the developers. This shared language is used to create models of real life processes in software.

- Our domain is basically **CALL** (*Computer-assisted language learning*) and more generally Computational Linguistics.
- Important papers: Nation (2014), Duolingo (2016)

Related research: quick summary

What is a word?

- A *word* is a very ambiguous term. For this reason other, more precise vocabulary is used by linguists.
- Are “run”, “running” and “ran” the same word?
- Who knows. They are different *tokens*. But they belong to the same *word family* and share the same *lemma*. However “run-” and “ran-” are two different *roots*.
- The Lomb model mostly uses the lemma as the fundamental learning unit. But sometimes (e.g. phrasal verbs) there is some additional logic. E.g. “put” and “put up with” should be different items.

How do people learn new words?

- Massive lack of consensus / models.
- SRS vs. 8-12 exposures.
- Key variables: frequency, spacing.
- Big debate: explicit (Duolingo) vs. implicit (Krashen) instruction.
- General lack of data.

The Duolingo Model for estimating PoR

- The Duolingo model is laughably simple:

$$p = 2^{-\Delta/h}$$

- Where Δ denotes the time since the last exposure and h is the *half-life* or a number which represents how good a person's memory of that item is. This is assumed to be:

$$\hat{h}_{\Theta} = 2^{\Theta \cdot x}$$

- And the weights Θ are found by gradient descent.

Deep Knowledge Tracing

- Deep Knowledge Tracing attempts to suggest a learning itinerary by taking in sequences of user interaction data (usually very sparse vectors where each exercise is a variable) and using the usual sequence learning techniques (RNNs, LSTMs, GRUs etc.).
- Mixed success, no one has explained why this model works.

Open research questions

- **Probability of recall problem.** Apply sequence learning to the PoR problem. Because Duolingo just aggregates all the data into a few variables much information is actually lost. We can use HLR and classical ML algorithms as benchmarks.
- **Book itineraries.** A user wants to read a book. But the book is too difficult and reading it is a slow, tortuous process. Can we suggest a sequence of books to build up his/her vocabulary?
- **Cold start problem.** A user might know many thousands of words. But currently we can only know this after the user has been on the app for a long time. Can we make an estimate of which words they know through CF after they read and interact with just a few pages of text?

Implementation Details

Summary of current development state

Work done: - Single-user app. - Minimal infrastructure and application layer. - Bounded contexts: - Tracking - Library - Revision - Tagging and translation - Reader - Problems - The content problem - Dataset

The content problem

One of the core idea of the app is to make it incredibly efficient for the user to find out what a word or sentence means. To do this, whenever a user clicks a sentence it must be translated and tagged. Doing this on-the-fly is slow and would scale terribly as the number of users grows. Furthermore, much computation would be wasted as books/texts get read again and again, and translated again.

A much better approach is to pre-translate and pre-tag texts before the user starts interacting with them. This presents a challenge, because not only is there a translation bottleneck of about 10 books/day, but translating and sharing books not in the public domain is illegal.

For this reason, the actual translation and tagging should happen outside of the app, which is just a platform for reading texts which are already translated and lemmatised. This presents no legality problem, (same as the Rio PMP300 case).

The solution: epubs

- The solution I have been working on recently is to modify epubs.
- The epub format is just a wrapper for html, css and images.
- Can be parsed and tags can be added, although doing this considering all the possible tags and edge cases would be very labourious and tedious.
- Such modified epubs can be read on a browser with a special reader I have been working on. They could potentially be read using low powered devices like ebook readers with some special software.

Dataset

Before I started working on modifying epubs directly, I would simply convert the epub to a text file and process / read /revise that. Ever since I wrote the tracking module I have been collecting data from my own interactions. I have been focusing on getting the epub reader working.