Labtwin challenge – answers to questions

Part 1

What other techniques would you apply to use these papers in the best way to train a transcription model?

First I'd like to run tests to make sure what I did is doing exactly what it is supposed to do.

Then I would like to change all numbers to spoken form (also those inside words), which I started doing but didn't finish before the 24 hour deadline.

The next step from what we have now would be to include special symbols and change them to spoken form (because they're important sometimes). Use UTF-8 or something, not plain ASCII and then to make a transformation that turns '°' into 'degrees', µ into 'micro', "it's" into 'it is" etc.

Include capital letters when they have meaning, such as in acronyms, since PCR and DNA are important words.

It would probably work better to have it context dependent, use some information from the environment the word appears in (the sentence it is in, the paragraph it is in, the paper it is in, what kind of paper it is).

I'd do some transformations and visualizations to get insight into what is contained in the papers, to research what to do. Use unsupervised clustering or use the keywords from the papers to sort them out and understand them. Maybe use the authors' fields of expertise?

Then it would be cool to have it context dependent, i.e. use some information from the environment the word appears in (the sentence it is in, the paragraph it is in, the paper it is in, what kind of paper it is).

And of course talk to someone with domain knowledge to find out what is important and what is not (and if my ideas make sense).

Part 2

How would you keep track of the overall transcription model performance, over time? Which KPIs are relevant and what kind of (acoustic) test data would you generate?

What is transcription?

Transcription is converting audio recordings of speech to text that people can understand.

What are the problems of transcription?

The transcriber can mishear and write the wrong word down. A perfect transcriber always writes the right words.

A common indicator for good speech recognition is 'word error rate' (WER), a variant of Levenstein edit distance, defined as

WER = (number of substitutions + number of deletions + number of insertions) / (number of substitutions + number of deletions +number of correct words).

Having worked as a transcriber myself I'm familiar with the human errors. The main cause for me was typos due to being in a hurry, and some spelling errors. This isn't a big problem for an artificial transcriber.

Sometimes I would hear the wrong word, but I understood the context so after a little confusion I figured it out. This is a bigger problem for artificial transcribers than people.

But is the transcriber solely responsible? Sometimes the user who speaks mispronounces it. There are two steps in the process; dictation (word to sound to recording) and transcription (recording to word to text.)

The WER could be compared to human transcribers and I guess the goal is to achieve at least the same rate as a good lab assistant and pass the Labtwin Turing Test.

Having test sets separate from the training sets is of course neccessary to evaluate performance and using something like cross-validation to validate the model.

Generating test data:

Using the best general transcription software to generate test data sounds like a good idea, but on second thought it doesn't seem to work so well because it isn't familiar with the specialized vocabulary. Having a person with domain knowledge and english skills oversee it would help.

Hiring people with some biology background and good typing and English skills to create the test data sounds pretty solid. Just make sure that there's noise in the background, to imitate real life situations.

Also make sure there's always test data for validation. Change the models. That is, Train / Test / Tune.

Additionally, how could user generated content be integrated into the model in order to improve the transcription accuracy? What sources of bias can occur and how would you deal with it?

People who use the transcriber generate data with their use. We'd need to separate the good from the bad (training on failed transcriptions would be detrimental).

We could have some kind of rating system, where the user can let us know if the transcription was good or not (with a written description of the semantic error).

Succesful user-generated transcriptions can be used as both training data and test data.

Factors that might cause bias that come into mind are:

Mother Tongue, Biological Field of Specialization,

English Proficiency Level, Education Level,

The circumstances which is recorded in

(open window? others talking?)

The recording device (though if it is an iPhone application then it is pretty standard, but could depend on which iPhone version it is or maybe its age)

maybe the user is wearing headphones with a microphone?

to name a few.

If users give their personal information, labelled data would be acquired that could be used to account for some of these problems, using methods from data science.