Mackenzie Myers

December 10th, 2021

DS 440

Final Report

Introduction

I was partnered with Jonathan Augustin for this capstone project. Our assigned project is to automate simplifying language in radiology reports to a sixth grade reading level. Our customers will be doctors from the Hershey Medical School, including Michael Goldenberg, Benjamin Shin, and Sunil Jeph. A radiology report is a document specifically written with complex anatomical terms for the review of healthcare providers. This makes them difficult for the average person to understand. Here is an example of what a radiology report may look like[1]:

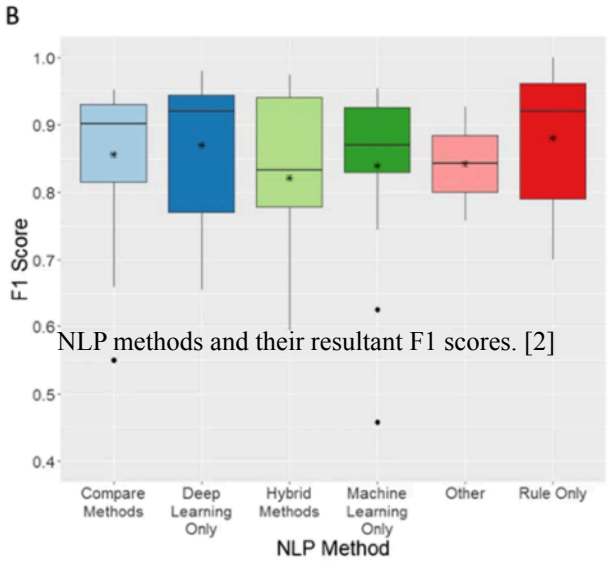
> There is marrow edema within the distal clavicle. There is a small AC joint effusion with mild pericapsular edema. No elevation of the distal clavicle and no fracture. The coracoclavicular ligaments are intact. There is a mild type II acromial configuration and minimal lateral downsloping of the acromion. No subacromial spurring.

By developing a method to translate these reports into simplified material, it will allow patients to understand what is being said.

Before we attempted to solve this problem, my partner and I first had to understand the problem. Upon being assigned this project our first step was researching radiology reports and what they look like. We found many resources online, the best being a website made by NationalRad which has many example reports. By finding examples of the reports we could see how the complex language would be an issue.

Our next step was to find out how much progress has been made by other researchers on this topic. We found several papers written with the same intent of simplifying radiology reports. One useful research paper written by Casey et al. describes which type of natural language processing was the most successful. They compared hybrid methods, deep learning only, compare methods, machine learning only, other, and rule only. Several approaches were used to develop testing, including k-fold cross-validation, and summed up in an F1 Score.

Graph 1



NLP methods and their resultant F1 scores. [2]

This was a great way for my partner and I to begin deciding which approach we would like to take when we begin developing our program.

As for work delegations, at the beginning of the semester, Jonathan's responsibility was to find the sample radiology reports and convert them into text files that will be useable as training data. Unfortunately finding medical reports is a difficult task because of the confidential nature of medical data. Jonathan found a few websites that have sample radiology reports, totaling about 25 reports. We would use these as our final testing data set.

My responsibility was to get our first model working and then decide where to improve. The code in question was Leveraging Social Media for Medical Text Simplification by Pattisapu et al [3]. This particular paper included a GitHub repository to the code they used. For the first half of the semester, I worked on this code to simplify medical text.

However, halfway through the semester I found that this code did not work as well as we would like, even after training it to over 100 epochs. The output sentences were non coherent and hardly retained any meaning from the original radiology report. I decided to switch paths and found a new, much more promising project to pursue. This second project was Paragraph-level Simplification of Medical Texts by Devaraj and Wallace of Cornell University [5]. This worked much better and gave me results much closer to what I was looking for.

Evaluation Method

In order to measure the readability of the radiology report, we will use the Flesch Kincaid readability score, as well as the Flesch Kincaid grade level score. These comprise of the following equations:

Flesch-Kincaid Readability =

$$0.39 \times (\text{Total Words} / \text{Total Sentences}) + 11.8 \times (\text{Total Syllables} / \text{Total Words}) - 15.59$$

This is the standard Flesch-Kincaid readability formula. A score of 0 reflects an extremely difficult document to read. A score of 100 is an easy document to read. Our target score is 70-80, which would be around a 6th or 7th grade reading level. The first exemplary fragment of text at the beginning of my report scored an 11.3, indicating it is an extremely difficult document to read and is at a college graduate level.

Flesch-Kincaid Grade Level =

FKRA = (0.39 x Average words/sentence) + (11.8 x Average syllables/word) - 15.59

The Flesch-Kincaid Grade Level approximates the grade level at which one could read the input sentences. For example, a grade level score of 5.8 would mean that a fifth or sixth grader could read the input.

Method 1: Medical Text Simplification by Pattisapu et al.

Pattisapu et al. uses a hybrid approach which uses both neural text simplification as well as lexical simplification, but also adds a denoising autoencoder. Social media is used train the

denoising autoencoder to overcome the limitations of the original hybrid models. In particular, they use medical blogs whose target audience is average people who are not doctors.

They first run MetaMap on the social media blog and select any post which contain at least three medical terms. The extract the sentences and split them into a training and testing set, with a corresponding ratio of 90% to 10%.

I experienced many difficulties getting this model to work on my machine. There was many roadblocks that I had to deal with. For one, you need a very particular software running in the background. It is called PyMetaMap which is a Python version of MetaMap. MetaMap is a program made by the National Library of Medicine that maps biomedical text to the UMLS Metathesaurus. To use MetaMap you need an UMLS license. On September 7th 2021 I received confirmation from the National Library of Medicine that they approved my license request and I was able to download the MetaMap file.

It took many errors, debugging and researching to figure out how to run the code on my machine. But after two weeks worth of trying to figure this out, I eventually was able to get the code to work in the way I wanted.

From there, I began training the model. I trained it up to 40 epochs at first to examine the results, and this is what the output was:

```
in october , he had a heart attack in the age of heart
failure .
they are also known as a person who have a person .
it is a heart attack on the age of the brain .
at the time , the united states has been used to treat a
heart attack in the time .
at the age of people , and was diagnosed with alzheimer 's
disease .
```

As you can see, these sentences do not make any sense. It is very clearly words strung together by a machine. At this time I decided to not yet run Flesch Kincaid readability scores on the output texts as their incoherence renders them useless anyway.

I thought that the reason the results are not as good as we would have hoped is because there is not enough training data as well as not enough trained epochs.

Therefore more first step was adding more data to the train set. The original raw data had text in three columns, with the first and second being unnecessary information for my study and the third column being the required sentences I wanted. I asked my partner Jonathan to help me clean this data to get just the desired sentences. I added the sentences to the end of aligned.en and aligned.sen according to wether it was simplified or not.

I trained this new expanded train set model up to 80 epochs and the results were still unfavorable. Here is example output:

```
influenza virus influenza a virus infects a virus .
the world virus is an infection of the virus .
They contain blood cells , which cause blood cells which
are found in humans and dendritic cells .
Some types of liver include breast cancer , brain cancer ,
leukemia , and lung cancer .
even patients who have been used to treat problems .
in the United States , from the world health organization
is used to be diagnosed with tuberculosis .
```

As you can see, the results are still incoherent. There is a lot of repeated phrases and words. If you review the input and output text, one can see how the output might be similar to the input, but the output has been so deformed that it has lost any real meaning.

It was at this time that I made the decision to look for other solutions to medical text simplification. I am glad I did so because it was then that I discovered a model that would work much better for our purposes.

Method 2: Paragraph Level Simplification of Medical Texts

I decided to explore other options for medical text simplification and found code that works very well on its own. It is written by Ashwin Devaraj of Austin University. Several pretrained models were included with the code which makes it much faster to be able to start using the models right away.

To exemplify just how much better this new code is, table 2 is the same example input from before, this time processed with the new model.

Table 2

Complex	Simplified
There is an intramedullary lesion of the fifth metacarpal shaft measuring approximately 2.4 cm in length . The appearance is most typical of an enchondroma and radiographic correlation is advised .	This is a small lesion on the fifth metacarpal shaft. This review shows that there is some evidence of an enchondroma, but the evidence is not conclusive. More research is needed.

Metamap is not used in this new model. Instead, it is trained on the Newsela dataset as well as a novel dataset called ‘Cochrane’ which was scraped from the Cochrane Database of Systematic Reviews. This database contains syntheses on a wide variety of medical topics. The critical component of this database is that each synthesis has a plain language summary (PLS) written by the author. 4459 pairs of these sections were mapped together and formed into the Cochrane dataset.

Example Excerpts of Complex vs. Simplified Synthesis in the Cochrane Dataset

> **Technical abstract:** Analysis showed a higher rate of weight gain in the high-volume feeds group: mean difference 6.20 g/kg/d (95% confidence interval 2.71 to 9.69). There was no increase in the risk of feed intolerance or necrotising enterocolitis with high-volume feeds, but 95% confidence intervals around these estimates were wide.
>
> **Plain-language summary:** Very low birth weight infants who receive more milk than standard volumes gain weight more quickly during their hospital stay. We found no evi- dence suggesting that giving infants high volumes of milk causes feeding or gut problems, but this finding is not certain.

An existing encoder-decoder model such as BART tends to prefer to delete words instead of paraphrasing, so this model fixes that by penalizing the decoder for production of technical tokens. This results in only a minor setback to content quality while dramatically boosting readability, which is what my project aims to do.

The paragraph level medical tech simplification includes four pretrained models: newsela, Cochrane, no-ul, and both. I have tested all four of the models to compare their differences and see if one stands out as being the best.

Before I introduce the comparisons of each model, I would like to talk about the evaluation metric I am using to decide which text was the most simplified. I chose the Flesch-Kincaid readability metric to measure 'simplicity' of text. I wrote a small python script that takes in text input and outputs the numeric Flesch Kincaid score. To do this I used the textstat package which includes a built in function to return the Flesch reading ease score and Flesch Kincaid grade level score. I will use this to compare the different models.

Table 3

	Flesch Kincaid Readability
Newsela	66.54
Cochrane	**70.94**
No UL	66.84
Both	66.74

From table 3 you can see that the model that used only the Cochrane dataset received the best readability score. Therefore I decided to focus on the Cochrane dataset as my main training dataset.

The next step was to improve this preexisting training model by adding the data Jonathan and I had collected to the train set to train the model on more data and therefore hopefully get more accurate results.

During the week of Thanksgiving break I trained a new model using the existing data from the paragraph level simplification code as well as previous data from the old model.

To evaluate the new model, I ran the old model as well as the new model on 5 example complex radiology reports. I then took the Flesch Kincaid reading grade level score of all of the outputs and took the average of the new model scores and old model scores. The average grade level scores are shown below:

	Flesch Kincaid Grade Level
New model	11.7
Old model	12.4

As you can see above, the new model performs marginally better than the old one.

One issue I have is that although the Flesch Kincaid score for the new model lowered slightly, this may be at sacrifice of the comprehension of the sentences. For example, see this example sentence pulled from the old and new model:

Old model

The brain shows evidence to have increased dural thickness, especially along its midline, but there is not enough evidence that it has changed significantly.

New Model

The brain shows signs that there is no evidence that there has been any brain damage to the brain.

From the above example outputs you can see the new model did simplify *more*, but did it simplify *better*? Some meaning of the sentence was lost. Instead of talking about the dural thickness of the brain the new model talks about brain damage. Depending on which doctor you

ask this might not be a correct translation. I infer that at some point the medical text may actually be over simplified. There could be a point where simplification loses original meaning of the text in translation.

The perfect simplification model would take this into account. The most ideal way to do this would be to have a doctor manually read tens or hundreds of sample simplified radiology reports from different models and pick the one that simplifies the best without losing the meaning of the original text. However, doctors are busy people so as you can see this is not such a realistic option, and a huge problem standing in the way of medical text simplification.

Final Thoughts and Conclusion

I learned many lessons throughout this process of developing a method to simplify medical text. The first is that if something is completely not working, and you know that it is unfixable at least within your scope of knowledge, it is not a bad thing to find a completely new method altogether.

Something else I learned, and not so much a single lesson but an entire process, is how to use data science to apply to a real world problem. I found this course somewhat challenging especially at the beginning because I had never done any machine learning before. I am taking it at the same time as I am taking CMPSC 448, which is the first machine learning course I have taken at Penn State. Not only did I have to quickly learn about what is machine learning and how does it work, but then I had to learn how do you actually use it to solve a real problem.

This was made a little easier since I was able to find pre existing code online where others before me have tried to simplify medical text. But it wasn't until the middle to end of the semester that I truly understood how machine learning works that I was able to start actually finding out what was happening with the deep learning 'black box.'

Another piece of knowledge I learned is how new the field of simplifying medical text in particular is, and the lack of resources available for it. Although the popular tool 'MetaMap' exists, which is somewhat useful, MetaMap has several issues with it that holds back medical text simplification. The first is that MetaMap does not truly translate complex terms into simple ones. It assigns levels of 'difficulty' to medical words and maps them together. The way most medical text simplification projects use it is by finding the word in MetaMap and then finding a word that is mapped to it that has an easier difficulty score.

MetaMap also has the downfall of being restricted use, meaning to even gain access to use MetaMap you need to create an account, submit a lengthy application and wait for one of their workers to manually approve you. After that you'll find that using MetaMap is a complicated process including starting a server that remote connects to the company that made MetaMap and then actually writing MetaMap into your code is an entire other set of steps. PyMetaMap is a custom python package available on GitHub written by AnthonyMRios which attempts to make this easier. He implements several python functions that make accessing the MetaMap information more streamlined. However, all of this seems overly complicated for a substitution process that should be as easy as looking up a term and finding its appropriate substitution.

I believe that a simpler thesaurus of medical text terms and substitutions should be made. If I had unlimited resources to do this project, I would have a team of doctors spending hours to be given a medical term and then enter a simpler term for it. I would map this thesaurus in a simple two column CSV, where you could search one column for the complex term and one column for the simplified version. Then the first step of medical text simplification could be to use this man-made dictionary to substitute all the complex terms for layman terms.

Overall, I am satisfied with my current model of medical text simplification, and proud of how well it turned out.

Sources

[1] “Diagnostic radiology reports,” *NationalRad*. [Online]. Available: https://nationalrad.com/radiology/reports/. [Accessed: 06-Sep-2021].

[2] Casey, A., Davidson, E., Poon, M. et al. A systematic review of natural language processing applied to radiology reports. BMC Med Inform Decis Mak 21, 179 (2021). https://doi.org/10.1186/s12911-021-01533-7

[3] Nikhil Pattisapu, Nishant Prabhu, Smriti Bhati, and Vasudeva Varma. 2020. Leveraging Social Media for Medical Text Simplification. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. Association for Computing Machinery, New York, NY, USA, 851–860. DOI: https://doi.org/10.1145/3397271.3401105

[4] Alan R Aronson. 2001. Effective mapping of biomedical text to the UMLS Metathe-saurus: the MetaMap program. In Proceedings of the AMIA Symposium. American Medical Informatics Association, 17.

[5] Devaraj, A., Marshall, I. J., Wallace, B. C., & Li, J. J. (2021, April 12). Paragraph-level Simplification of Medical Texts. Cornell University. Retrieved November 10, 2021, from https://arxiv.org/abs/2104.05767.