USCid:2518534934 Name:Po-hsuan Yang

Report

Tika is a great tool I have never used that before. At first, I thought Tika can only parse “text” in PDF. But, finally I found that it can parse “image” for text. It is really cool, I can parse text in image. I learned a lot from Tika through websites. I found many websites parse PDF to text are all using Tika. I think Tika is easy to use. I use PDFParser.parse to get all content out. Some content couldn’t be parsed out because the original PDF files have so much noise even I don’t know what the original words are.

It’s a cool project. I use regular expression ("\\b"+keyword+"\\b") to find keywords. I add white spaces at the front and at the end of a keyword because some words can affect the results. For example, “disc” is the prefix of “discrimination”. Without white spaces, I would get many irrelevant results. Also, I use case insensitive because maybe some keywords have Upper-case at the first character.

Finally, I found that there are 231 file that contain keywords. More than 900 keywords occur in these files. But some files are not exactly talking about UFO. Take a keyword “disc” which teacher provides as an example. They are not taking about UFO because only appears once in that document. Maybe it’s about disc for computer. Or some content have keywords but we don’t parse them out because of so much noise in document, even I don’t know what the words are. Or Tika parser couldn’t parse correctly for some words. For example, “sau?er” I think it is “saucer”, but we cannot parse it out. From my observation, all these reasons I mentioned could affect my results. By using Levensthein algorithm for extra credit, I think I can improve some results.

Extra Credit Report

I write Stopwords.java which is a class to remove stop-words. I just got the stop-words list from the website. And I got Levensthein algorithm from the website.

It’s a interesting project. I did two big revise for my program. First version, I remove punctuation. And also replace escape characters into white space. Then chop them by white space. So, I get an array of string. For each keyword and each content word, I run Levensthein algorithm to find the distance between these words. I assume that if the distance is ONE, then we can say the original document miss-typed or the document has noise and Tika couldn’t parse it correctly. For example, “sau?er” I think it is “saucer”.

They are many words that distance is one from my keywords list. For example, a keyword “craft” and I found that I parsed many files contain “draft”. And “UFO” could be “WFO” or “HFO” or “UF”. So I found that many files don’t contain keywords I wanted. Also, I found that if I chop by white space, I couldn’t find the keyword which is bi-words. Then I did second version.

Second version, I also remove punctuation, replace escape characters into white space and chop them by white space. Learned from the first version, I assume there must have more than one keyword cannot be miss-typed. So, if a document contains exactly one of the same keywords, I can say that this document is talking about UFO. Then, I run Levensthein algorithm to find which words in document are miss-typed. Also, I considered bi-words. I did 2-gram to find phrase for keywords.

Finally, I find that by using Levensthein algorithm a keyword “black aircraft” can be find twice but we cannot find any result without using this algorithm. And the keyword “UFO” has a huge different because UFO is only three characters, so many words can have only one distance from UFO. For example, “WFO” or “HFO” or “UF”. But these documents are truly talking about UFO. So, that’s the main reason why I make an assumption at second version that there must have more than one cannot be miss-typed.