International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 3 (2015) pp. 7721-7726
© Research India Publications
http://www.ripublication.com

Rule Based Chunk Extraction from PDF Documents Using Regular Expressions and Natural Language Processing

Amol Rajaram Karad[a] and Rahul Raghvendra Joshi[b]

[a,b]Computer Science Department,
Symbiosis Institute of Technology (SIT),
Affiliated to Symbiosis International University (SIU),
Gram-Lavale, Tal-Mulshi,
Pune, 412115, Maharashtra, INDIA.
E-mail (s): amol.karad@sitpune.edu.in,rahulj@sitpune.edu.in

Abstract

The Natural Language Processing (NLP) is a stimulating and vital field of Artificial Intelligence (AI). The NLP can be used to find out the required intelligence through the system under consideration, so that system behaves as per convenience and efficiency expected by the user. The proposed system demonstrates application of NLP and by using Regular Expressions to categorize and classify sentences in Word/ PDF (Portable Document Format) documents according to rules provided by user. Thousands of similar kind of PDF documents can be easily processed by reading them page wise, the proposed system produces results according to the user defined rules those are applicable to all input PDF documents. Single rule is written by considering one input PDF document and apply the same to all other input PDF documents of the proposed system to create individual data chunks out of all documents and display them on User Interface in table format.

Keywords: Natural Language Processing (NLP), Regular Expressions, Artificial Intelligence,PDF, data chunks, User Interface

1. Introduction

Natural Language Processing (NLP) is an area of research and application that explores how computers can be used to understand and manipulate natural language text or speech to do useful things. The motivation behind NLP is to make computers learn human beings language rather than one’s learn theirs. Text is the largest repository of human knowledge and is growing quickly: Legal documents, emails, news articles, web pages, transcripts of phone calls, technical documents, government documents, patent portfolios etc. Current paper is mirror image of rule based processing approach [9]. NLP researchers aim to gather knowledge on how human beings understand and use language so that appropriate tools and techniques can be developed to make computer systems understand and manipulate natural languages to perform the desired tasks [4]. When intelligence came in to picture NLP plays a vital role in terms of speech recognition [8], question answering system [8] etc. and continuous research work is going on in these fields. Since in today's world the data produces on internet is in huge quantity, and surely it's not feasible to process, manipulate and get required part from produced data through traditional methods. [2], user deserves to automate this kind of work. Proposed system provides such a flexibility to process all the legal PDF documents and extract the important part from the document using NLP and Regular Expressions [6].

2. Related Work

Artificial Intelligence (AI) is an important aspect in order to deal with text or speech processing. One can work on computing with words (CWW) [1] however working with natural language and ignoring AI is not sounds good. It is feasible to process the data from web pages using rule based system [7]. The translation is having language problem like Chinese, Spanish, and English. Some available analysis engine supports only in Chinese language.[2] To assess quality factors in documenting the source code NLP is used however to process same kind of tremendous documents system is not feasible[3]. One can do the Information Extraction [4] from structured documents since it is difficult to extract same by using single rule file. Given system is having the strength to do with single principle file.

3. Architecture of Proposed System

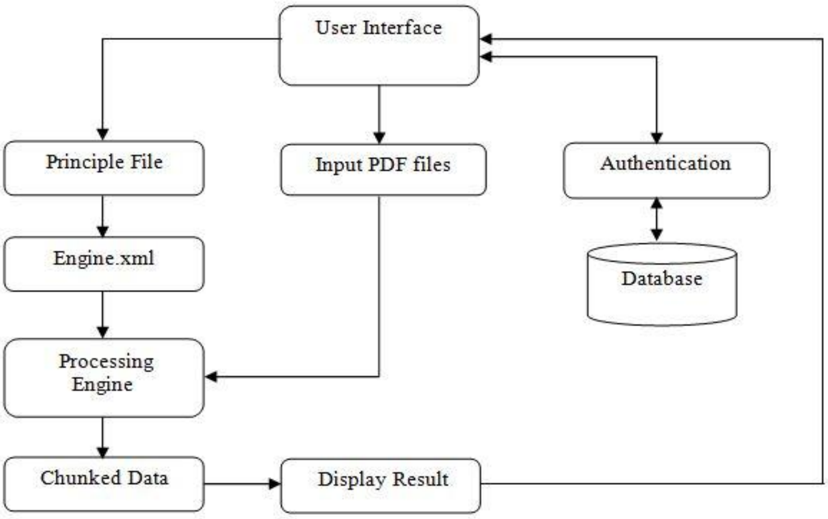


Figure 1: Architecture of proposed system

4. Proposed Work

a. User Interface

User registers and login into the system through userinterface (UI). The records and logs of registered users are stored in the proposed systems database. After login into the system, user can access the respective input PDF document/documents which are going to process. The UI also facilitates to browse the principle file which can be used to process the input PDF document. After the completion of processing of PDF, UI displays the output in terms of table format in which last column from right hand side contains actual chunked data of required parts from the PDF documents.

b. Methodology used

Main logic behind the proposed system is shown in Figure 2. The Engine.xml file can contain one to ‘n’ number of attributes and also, for each attribute one to ‘n’ number of rules can be framed. The rules contain annotations like first section of document, first subsection within that respective section, start word or phrase within that subsection and End of Paragraph (EOP) within same subsection.

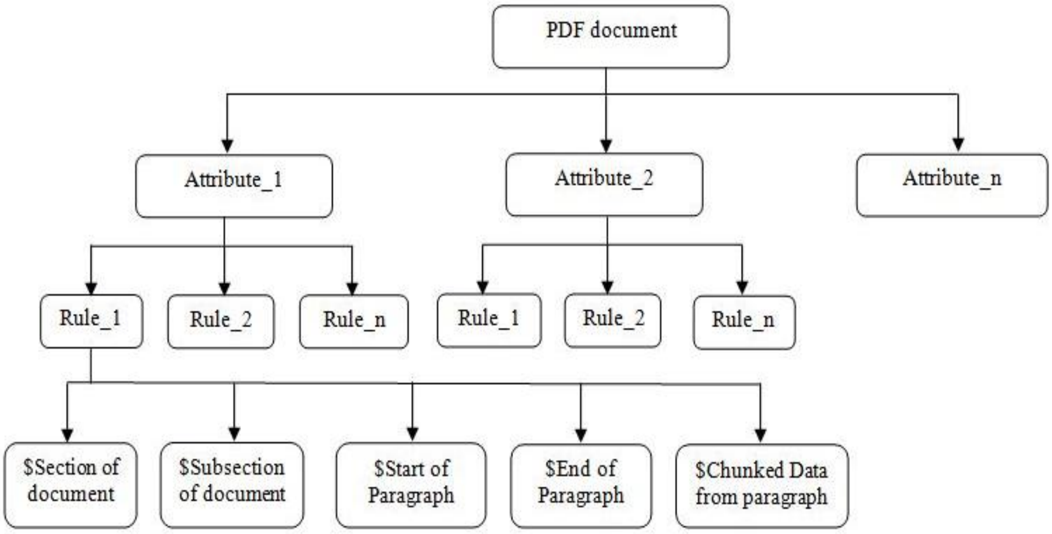


Figure 2: Tree structure representing annotations of rules applied to particular section of the document

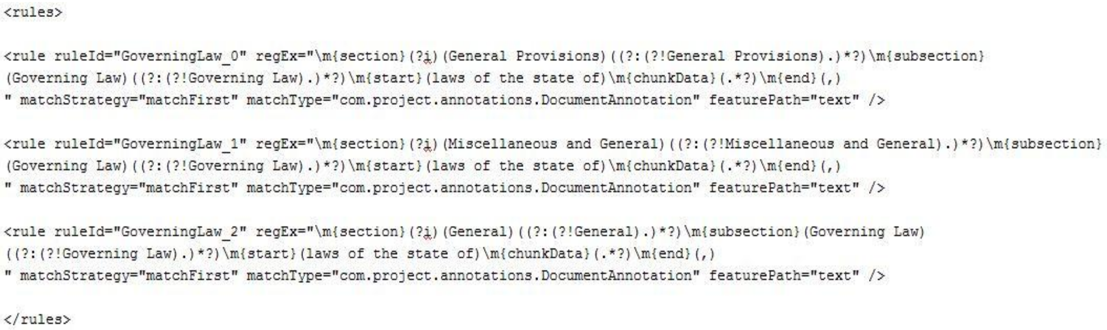
c. Principle File

Principle file is the simple text file and is key part of the system. User can write rules according to one’s requirements .User first decide what himself/herself expects from kind of input documents which are under process? The answer of this question is nothing but the principle file. Since to process all PDF documents those are having same format in terms of section, subsection, however varying their relative values within the same, only one principle file will be written by user and same will be applied to all the input files.

The general structure of the file is as follows:
USER($IGNORE_CASE){$RULEID("*Rule_Id*")$SECTION("*title of the section*")$SUBSECTION("*title of the subsection*") $START("*start of the paragraph*")$END(*EOP*)}

The structure contains IGNORE_CASE parameter which means if the documents are both in upper case or lower case it ignore the case sensitivity while matching the rule. SECTION and SUBSECTION parameters match the first word in the documents, if any one of these is missing, it will skip and directly find START and End of Paragraph (EOP).The contents present in between START and EOP will be chunked by the system.

One single rule written with 3 ruleId and generated regEx is shown in Figure 3. Now same rule will be applied to all input PDF files and many rules of such type can be written for all input PDF documents.



```
<rules>

<rule ruleId="GoverningLaw_0" regEx="\m{section}(?i)(General Provisions)((?:(?!General Provisions).)*?)\m{subsection}
(Governing Law)((?:(?!Governing Law).)*?)\m{start}(laws of the state of)\m{chunkData}(.*?)\m{end}(,)
" matchStrategy="matchFirst" matchType="com.project.annotations.DocumentAnnotation" featurePath="text" />

<rule ruleId="GoverningLaw_1" regEx="\m{section}(?i)(Miscellaneous and General)((?:(?!Miscellaneous and General).)*?)\m{subsection}
(Governing Law)((?:(?!Governing Law).)*?)\m{start}(laws of the state of)\m{chunkData}(.*?)\m{end}(,)
" matchStrategy="matchFirst" matchType="com.project.annotations.DocumentAnnotation" featurePath="text" />

<rule ruleId="GoverningLaw_2" regEx="\m{section}(?i)(General)((?:(?!General).)*?)\m{subsection}(Governing Law)
((?:(?!Governing Law).)*?)\m{start}(laws of the state of)\m{chunkData}(.*?)\m{end}(,)
" matchStrategy="matchFirst" matchType="com.project.annotations.DocumentAnnotation" featurePath="text" />

</rules>
```

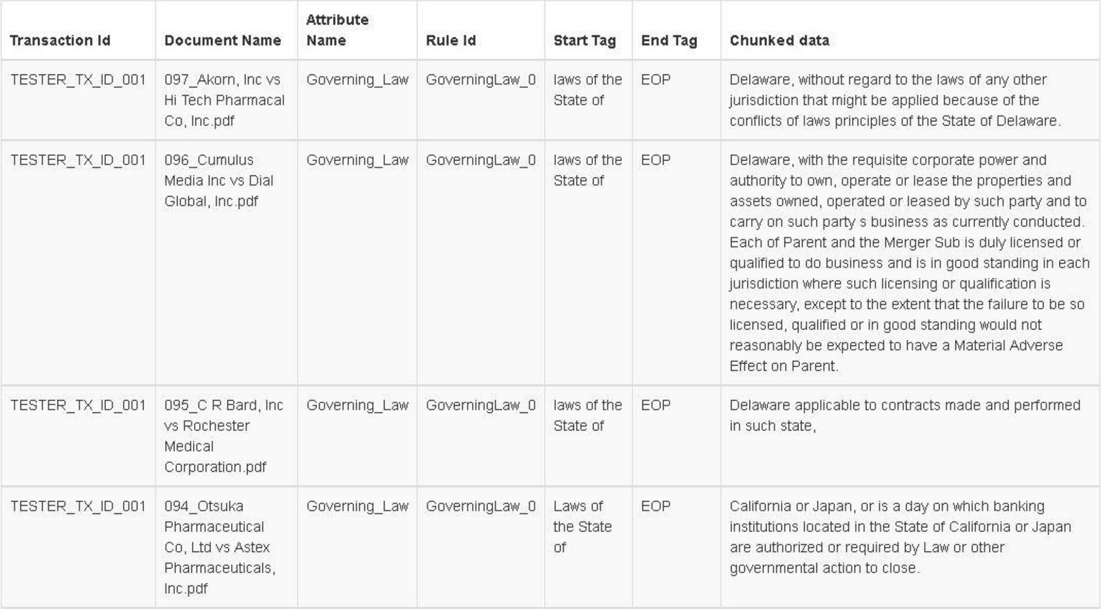
Figure 3: Engine.xml file

5. Results of Proposed System

The results contain seven different columns. The first column of the result is Transaction Id which is unique for each user. The second column contains name of the PDF document which is to be processed. The third column contains name of the attribute and fourth column contains Rule Id, there can be one or more number of rules will be farmed for a single attribute. The fifth column contains Start tag indicating starting word or phrase of chunked data. The sixth column contains End tag indicating endword/phrase or end of paragraph of chunked data and the last column contains actual chunked data extracted from respective input PDF document/documents.

For instance the results contains first attribute as 'Governing_Law', which is a keyword for input PDF file. Under this 'Governing_Law' attribute rules are framed which may contain 1 to 'n' number of Rule. The first Rule Id is 'GoverningLaw_0', similarly, second Rule Id is 'GoverningLaw_1', third Rule Id is 'GoverningLaw_3' and so on (as shown in figure 3 of Engine.xml file). By considering only first Rule Id i.e. 'GoverningLaw_0' (As shown in Figure 3 of Engine.xml file) results are shown in Figure 4 of proposed system. Results shown here of the proposed system contains four PDF documents as input. User may give thousands of PDF documents according to his convenience as input. Full results including next Rule Id ‘GoverningLaw_1’ and further are just next table on UI after results shown below which are of similar type as shown in Figure 3. For sake of simplicity results are shown only for Rule Id ‘GoverningLaw_0’. So by applying attribute name as ‘Governing_Law’, Rule Id as ’GoverningLaw_0’, Start tag as ‘laws of the state of’ and end tag as ‘EOP’ to these input PDF documents, required data in chunked form is extracted.

Likewise Rule Id ‘GoverningLaw_1’ can be used ,same like attribute name as ‘Governing_Law’ other attributes can be used with the rules to extract other required details through supplied input PDF/PDF’s in chunked format.



Transaction Id	Document Name	Attribute Name	Rule Id	Start Tag	End Tag	Chunked data
TESTER_TX_ID_001	097_Akorn, Inc vs Hi Tech Pharmacal Co, Inc.pdf	Governing_Law	GoverningLaw_0	laws of the State of	EOP	Delaware, without regard to the laws of any other jurisdiction that might be applied because of the conflicts of laws principles of the State of Delaware.
TESTER_TX_ID_001	096_Cumulus Media Inc vs Dial Global, Inc.pdf	Governing_Law	GoverningLaw_0	laws of the State of	EOP	Delaware, with the requisite corporate power and authority to own, operate or lease the properties and assets owned, operated or leased by such party and to carry on such party s business as currently conducted. Each of Parent and the Merger Sub is duly licensed or qualified to do business and is in good standing in each jurisdiction where such licensing or qualification is necessary, except to the extent that the failure to be so licensed, qualified or in good standing would not reasonably be expected to have a Material Adverse Effect on Parent.
TESTER_TX_ID_001	095_C R Bard, Inc vs Rochester Medical Corporation.pdf	Governing_Law	GoverningLaw_0	laws of the State of	EOP	Delaware applicable to contracts made and performed in such state,
TESTER_TX_ID_001	094_Otsuka Pharmaceutical Co, Ltd vs Astex Pharmaceuticals, Inc.pdf	Governing_Law	GoverningLaw_0	Laws of the State of	EOP	California or Japan, or is a day on which banking institutions located in the State of California or Japan are authorized or required by Law or other governmental action to close.

Figure 4: Results of the proposed system

6. Future Scope

The more text processing in the form of chunked data is possible on PDF documents. The natural language processing tools like General Architecture for Text Engineering (GATE) developer [10] can be used to find annotations like date, location and name of the organization etc., by supplying the resultant chunked data as an input to GATE tool. This will be used to extract required parts through supplied input PDF documents. So, by using this user can directly find out required annotations through GATE tool.

7. Conclusion

In order to process all similar kind of PDF documents, proposed system reduces hectic workload of reading each document manually line by line and find out important paragraph. Proposed Natural Language Text Processing system automates the process and provides desired parts from each documents instantly, accurately and in a systematic way.

References

[1] Janusz Kacprzyk,Sławomir Zadrozny. Computing With Words Is an Implementable Paradigm: Fuzzy Queries, Linguistic Data Summaries, and Natural-Language Generation. IEEE TRANSACTIONS ON FUZZY SYSTEMS: VOL. 18, NO. 3, JUNE 2010

[2] XiaoguangYue, Guangzhi Di, Yueyun Yu, Wei Wang, Huankai Shi. *Analysis of the Combination of Natural Language Processing and Search Engine Technology*. 2012 International Workshop on Information and Electronics Engineering (IWIEE); Procedia Engineering: 29(2012)1636-1639.

[3] NinusKhamis, JuergenRilling, René Witte. Assessing the quality factors found in in-line documentation written in natural language: The JavadocMiner. Data & Knowledge Engineering: 87 (2013) 19–40.

[4] KaihongLiu, William R. Hogan, Rebecca S. Crowley.*Natural Language Processing methods and systems for biomedical ontology learning.*Journal of Biomedical Informatics*:* 44 (2011):163–179.

[5] Trevor Martin, *Member, IEEE*, Yun Shen, and Ben Azvine. *Incremental Evolution of Fuzzy Grammar Fragments to Enhance Instance Matching and Text Mining*. IEEE TRANSACTIONS ON FUZZY SYSTEMS: VOL. 16, NO. 6, DECEMBER 2008.

[6] Wei Lin, Yi Tang, Bin Liu, Derek Pao, XiaoFei Wang. *Compact DFA Structure for Multiple Regular Expressions Matching*; IEEE ICC 2009 proceedings.

[7] Dr. Y. V. Haribhakta,BhavikaChheda,NupurAgrawal, ShrutikaGirme.*Literature Survey in Natural Language Processing In the Sphere of Relation Extraction*. The International Journal Of Engineering And Science (IJES):Volume 3;Issue 7;2014;Pages 01-08 .

[8] Tsukasa Sagara,Masafumi Hagiwara. Natural language neural network and its application to question-answering system. Neuro computing 142 (2014);201–208.

[9] Erik Cambria, Bebo White. Jumping NLP Curves: *A Review of NaturalLanguage Processing Research*; IEEE Computational intelligence magazine ;may 2014; *Date of publication:11 April 2014.*

[10] Nancy Ide, Keith Suderman. Bridging the Gaps: Interoperability for GrAF, GATE, and UIMA.