Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 4 (2012) 365 – 369

Procedia Technology

C3IT-2012

Information Extraction from the Un-Structured Document using Grammatical Inference and Alignment Similarity

Ramesh Thakur[a], Suresh Jain[b], Narendra S.Chaudhari[c], Rahul Singhai[a]

[a]International Institute of Professional Studies, DAVV, Indore, India
[b]KCB Technical Academyy, Indore, India
[c]Indian Institute of Technology, Indore, India

Abstract

Huge amount of information is available in un-structured (text) documents. Knowledge discovery in un-structured document has been recognized as promising task in the recent years. Since un-structured document is typically formatted for human viewing, it varies widely from document to document. Frequent changes made to their formatting further causes difficulty in construction of a global schema. So, Discovery of interesting rules form it is complex and tedious process. Most of the existing system uses hand-coded wrappers to extract information, which is monotonous and time consuming. In this paper we propose a novel and hybrid approach of learning (context-free) grammar rules that are based on alignment between texts. Also it automatically discovers the grammar rules using grammatical inference of repeated pattern present in un-structured (text) document. The generated rules can be used to infer the attribute value pairs from the unstructured text document.

Keywords：*Alignment profile; Learning systems; Knowledge discovery; Sequence Mining; Grammar Inference;*

1. Introduction

Learning grammatical information from given sample of texts has attracted a lot of attention in past few decades. Application include computational linguistics [12]-[14], bio-informatics [15], [16], structural pattern recognition [17]-[19] and compression [20] [21]. Extracting information from text document is usually done by software called wrappers. Information is mined through the approach used in subheadings, images and formulae. Wrapping text document is usually based on manual technique [1, 2, and 3]. The problem with manually coded wrappers is that writing them is difficult and time-consuming job, as there is no standard semantics of text document. As a result the key challenges, for un-structured document information extraction is to develop the technique that allows the automation of extraction process. In automated grammar learning, the task is to infer grammar rules from given sentences of the target language. The sentences (strings of alphabet) are given as example for grammar learning.

doi:10.1016/j.protcy.2012.05.056

We have constricted on un-structured data source containing data, which is fielded but not tapered by global schema. Document such as address book, product catalog, and classified advertisement listing etc falls into this categories. Our work uses the alignment between the texts of a given collection of sentences and grammatical inference to automate the construction of wrappers to facilitate the process of information extraction.

2. Context-free Grammar Inference

The problem of learning the correct grammar from finite example of the unknown language is known as grammatical inference problem. A good survey of the field is due to Lillian Lee [4] and colin de la Higirea [5]. Although, context free language is well understood, there are serious limitations to learning them. According to Gold leaning from text is much harder problem [6]. Effective learning of formal language is another challenging research problem. The formal model provides an operational framework for numerous practical applications. It is useful in data mining, information extraction and prediction [7,8,9] In the case of sequence mining, datasets normally have only positive example and not negative example, so we are interested in inferring CFG form positive example data. In fact Gold [10] shows that not all language can be inferred from positive example only. A language that can be inferred by looking at a finite number of positive example can only said to be *inferable in limit* [10]. The grammatical inference given by Gold[10] introduced the notion of identification in the limit. This notion concerned with limiting behavior of an inference algorithm on an infinite sequence of example. Formally, a complete presentation of language L is an infinite sequence of ordered pair (w,l) from $\sum^* \times \{0,1\}$ where l=1 if $l \in L$ and 0 otherwise and every string $w \in \sum^*$ appears at least once. If an inference method M is run on larger and larger initial segments of a complete presentation, it will generate a infinite sequence of guesses g_1, g_2,g_3, etc. M is said to identity L in the limit if there exist some number m shut that all of the guesses g_i are the same for $i \geq m$, and g_m is equivalent to L is an infinite. This approach is not directly feasible for the web document task, science only positive set of example is available. Gold showed that any class of languages containing all the finite language and at least one infinite language couldn't be identified in the limit from only positive sample. As a consequence, the large body of researchers works on it. This research has led to identification of several such classes [11].

3. Grammatical Inference algorithm

Alignment based learning (ABL) [22] is based on alignment information. In ABL. Pair wise alignment for each pair of the input sentences is done by finding equal parts and unequal parts. Pair wise alignment is an arrangement of two sequences, which shows where the two sequences are similar and where they differ. A good alignment shows the most significant similarities, and the least differences. The algorithm considers the relative complexity of candidate grammar. For convenience we take the hypothesis to the set of stochastic context free grammars. Stochastic context free grammar is the context free grammar with probabilities attached to their productions. This augmented space is more continuous than the space of standard context-free grammars and provide more freedom for modifying candidate grammars. For example, the production $X \to u$ can be continuously deformed into1 the production $X \to u \mid w$ by varying the probabilities of w alternative from 0 to 0.5. The probabilities also make it possible to quantitatively assess the fit between a candidate grammar and language sample by calculating the probability that the given grammar would have generated.

We split the problem of grammatical inference into following phases:

- Codification of string: Before we can apply this algorithm, we need to transform the data into suitable format. The algorithm expects a set of positive sequence of confiscating of symbols from a finite alphabet set based on their alignment. So the strings (sentences) of input data sets are codified based on their syntactic categories.

- Calculate probabilities: for all sub string calculate the probabilities $w_1, | w_2, | w_3 \ldots\ldots | w_n$ $[P_1, P_2, P_3, \ldots\ldots, P_n]$ where $w_1, w_2, w_3 \ldots\ldots w_n$ are string occurring in the sample and $P_1, P_2, P_3, \ldots\ldots, P_n$ are their relative frequencies. If all string are different, then the P_i will be equal to 1/m where m is number of string how ever the p_i may very if some string appear more than once in the sample.
- Discovery of pattern: Searching of repeated sub-string‘s’ are performed and the sub-string‘s’ that occured multiple times are indicated by its associated probabilities, a new grammar $Y \rightarrow s$ are added and all occurrences of s are replaced by Y.
- Multiple Production alternative: If the occurrence of *s* is in such a position that multiple production alternatives are possible $(X \rightarrow us | ws)$ then new production is $Y \rightarrow u | w$ and $X \rightarrow Y's$
- Redundant production: Merge redundant rules and drop production, which are inaccessible (cannot be reached from start symbol).

3.1. Algorithm

```
Input: A corpora C of flat sentence (Codified string). Max_number(substring)
   Output: Set of CFG rules R
Begin
    Initialize rule set $R = \phi$
        // Calculate sub-sub-strings
        while $\forall \alpha \in C, |\alpha| > 1$ do
                $\Sigma$= sub-string(C)
                for each sub-string $\beta \in \Sigma$ do        // calculate the relative frequencies of  sub-string $\beta$
                $P_\beta$= calculate_relative_frequency$(\beta, \Sigma)$ // This procedure return probability  for the sub string $u$ in the corpora.
                end for
            $\gamma$=select $\beta$ of highest relative_friquency $P_\beta$
                    N= select next non terminal symbol // add new rule to rule set R
    if $|\gamma| > 1$ then

                if $\{\gamma \neq us | ws\}$ then

    $R = R \cup \{N \rightarrow \gamma\}$ // apply replacement rule for each string in the corpora

                update $(C, N \rightarrow \gamma)$
    else // $\gamma$ has multiple production alternative.

                $R = R \cup \{N \rightarrow Y's\}$
                Update$(C, N \rightarrow Y's)$
                N= select next non terminal symbol
                $R = R \cup \{N \rightarrow u | w\}$
                Update$(C, N \rightarrow u \text{ or } N \rightarrow w)$
        end if
  end if
end while // drop redundant alternative.
                $\forall X \in R$ remove multiple occurrence $X \rightarrow s$ preserving one occurrence

end.
```

3.2. Experimental Results

The proposed algorithm has been implemented using a code written in C programming language. The main data structures are stored in array of character and for sub string and relative frequency count the array of structure (string and double field) are used respectively. It uses the alignment between the texts of a given collection of sentences and probabilistic relative frequency for replacement rule in the corpora.

Table 1. *The detail of Nokia Mobile (A part of text file for Mobile Listing)*

Site address	Model	Price	Description
nokia-e7_1146.html	Nokia E7	Rs.25691	A QWERTY plus .....
nokia-n900_897.html	Nokia N900	Rs.23529	Nokia N900 is a high .
nokia-n97-mini_898.html	Nokia N97 Mini	Rs.17788	12 MP Camera …..
nokia-x6-16gb_1080.html	Nokia X6 16GB	Rs. 330,00	Auto Exposure, Auto

3.3. Codification of String

The main job of codification is to replace the string by a token so that the further processing becomes simple. It considers the alignment and font size. A string having same alignment is replaced by same tokens.

S_1: T1, S_2: T2 T3 T4 T5, S_3: T6 T7 T8 T9, S_4: T6 T7 T8 T9, S_5: T6 T7 T8 T9, S_6 : T6 T7 T8 T9

Where s_i is the string after token identification

Table 2. *Coded text of Table 1(Discovery of patterns for the context-free languages)*

Iterations

Y→T6 T7 T8 T9	Z→T2T3T4T5	X→T1 Z YYYY
T1	T1	
T2 T3 T4 T5	Z	
YYYY	YYYY	

3.4. Output

After applying the proposed algorithm the following set of grammar results:

X→T1 Z YYYY , Y→T6 T7 T3 T9, Z→ T2T3T4T5 | ε

We can interpret this as fallows: the start symbol represents a complete text. We can see that a text begins with fixed preamble; followed by a variable number of occurrences of Y each represents a single listing. A listing consists of a raw, which contains reference to their text Z. The non-terminal Y corresponding to a listing we may generate wrappers that segment each listing by searching the pattern specified by Y. The data fields for each listing can be extracted by mapping the text symbols to their actual content. Then the domain specific heuristics can be used to identify the semantic meaning of different fields. Science domain specific knowledge is not used in grammar generation it is used in the last step so this approach can be easily used in other domain. After applying the procedure we get following attribute value.

Table 3. Partial listing of extracted records

Site address	Model	Price	Description
nokia-e7_1146.html	Nokia E7	Rs.25691	A QWERTY plus .....
nokia-n900_897.html	Nokia N900	Rs.23529	Nokia N900 is a high .
nokia-n97-mini_898.html	Nokia N97 Mini	Rs.17788	12 MP Camera …..
nokia-x6-16gb_1080.html	Nokia X6 16GB	Rs. 330,00	Auto Exposure, Auto

4. Conclusion

This paper describes a general approach for generating information extraction wrappers using grammatical inference that enables information extraction from the un-structured document. In this work we have extracted attribute value of frequently occurring data from data intensive document. This work can be seen as a component of the larger goal of extracting knowledge repositories in un-structured text.

References

1. Atzeni P., Andmecca G. Automatic information extraction from large web sites. In.: Proceedings of the 16th ACMSIGMOD International Symposium on Principles of Database Systems (PODS'97) (Tucson, AZ). ACM, New York; 1977, p. 144–153.
2. HAMMER J., GARCIA-MOLINA H., CHO J., ARANHA R, CRESPO A. Extracting semistructured information from the Web. In Proceedings of the Workshop on the Management of Semistructured Data (in conjunction with ACM SIGMOD 1997). ACM, New York; 1997.
3. SAHUGUET A., AZAVANT F. Web ecology: Recycling HTML pages as XML documents using W4F. In Proceedings of the 2nd Workshop on the Web and Databases (WebDB'99) (in conjunctionwith SIGMOD'99). ACM, New York. 1999.
4. Lillian Lee. *Learning of context-free languages: a survey of the literature*, Report TR-12-96, Center for Research in Computing Technology, Harvard University, 1996.
5. Higuera C.D. L*Current trends in grammatical inference,* in Proc.Joint IAPR Int. Workshops Adv. Pattern Recognit, p.28–31, 2000.
6. Gold E. M., Complexity of automaton identification from given data, Informationand Control 1978;37: 302-320.
7. Giles, C., Lawrence, S., Tsoi, A.C.: Noisy Time Series Prediction using Recurrent Neural Networks and Grammatical Inference. Machine Learning, 2001; 44: 161-184.
8. Freitag D. *Using grammatical inference to improve precision in information extraction*, In Proc. ICML-97Workshop on Automation Induction, Grammatical Inference, and Language Acquisition, Nashville, TN, Morgan Kaufmann,San , CA, 1997.
9. Hingston, P. *Using finite state automata for sequence mining.* In Proceedings of the 25th Australasian conference on computer science, Australian Computer Society, p. 105–110, 2002.
10. GOLD, E. M.. Language identification in the limit. Inf. Cont. 1967, 10, 5**:** 447–474.
11. ANGLUIN, D. Inference of reversible languages. J. ACM 1982.,29, 3**:** 741–765.
12. P. Adriaans, M. Vervoort .*The EMILE 4.1 grammar induction toolbox,* Proc. ICGI, vol. 2484, p.293 , 2002.
13. M. Mohri . Finite-state transducers in language and speech processing, Comput. Linguist., 1977, 23: p.269 .
14. M. Mohri , F. C. N. Pereira,M. Riley. The design principles of a weighted finite-state transducer library, Theor. Comput. Sci., 231**:** p.17 , 2000.
15. A. Brazma , I. Jonassen , J. Vilo, E. Ukkonen. *Pattern discovery in biosequences,* Proc. ICGI, Springer vol. 1433, p.25 , 1998.
16. I. Salvador, J. M. Benedi . RNA modeling by combining stochastic context-free grammars and n-gram models, Int. J. Pattern Recogn. Artif. Intell.,2002, 16: p.306.
17. K. S. Fu , T. L. Booth. Grammatical inference: Introduction and survey, IEEE Trans. Syst., Man, Cybern.,1975, 5: p.59.
18. L. Micket. Structural Methods in Pattern Recognition, 1986.
19. S. Lucas , E. Vidal , A. Amari , S. Hanlon, J. C. Amengual. In Proc. ICGI, Springer 862, p.168 , 1994.
20. C. Nevil-Manning, I. Witten. Identifying hierarchical structure in sequences: A linear-time algorithm, J. Artif. Intell. Res., 1997, 7: p.67.
21. J. R. Rico-Juan , J. Calera-Rubio, R.C. Carrasco. *Stochastic k-testable tree languages and applications*, Proc. ICGI, Springer, 2484: p.199 , 2002.
22. M. V. Zaanen. *Implementing alignment-based learning,* in Proc. ICGI(Lecture Notes in Computer Science), 2484:, pp. 312–314,2002.