International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

Document Annotation with Caption Generating

Jisha James[1], Meenu Varghese[2]

[1]Computer Science and Engineering, ICET, Muvattupuzha, India

[2]Assistant Professor, Information Technology, ICET, Muvattupuzha, India

Abstract: ***Collections of huge, large textual data contains significant amount of structured information, which remains hidden in unstructured text. Relevant information is always difficult to find in these documents. In this paper we proposed an alternative approach that facilitates the generation of the structured metadata by identifying documents that are likely to contain information of interest and this information is going to be useful for querying the database. Here people will likely to assign metadata related to documents which they upload which will easily help the users in retrieving the documents. In this paper, a method of sentence generation from a given text. The task of sentence generation has its usage in text summarization. The technique that we have applied is N-gram language model.***

Keywords: Information Extraction, N-gram, Annotation, Sentence Generation

1. Introduction

There are many application domains where users create and share information; for instance, news blogs, scientific networks, social networking groups, or disaster management networks. Current information sharing tools, like content management software (e.g., Microsoft SharePoint), allow users to share documents and annotate (tag) them in an ad-hoc way. Similarly, Google Base allows users to define attributes for their objects or choose from predefined templates. This annotation process can facilitate subsequent information discovery.

Many systems do not have the basic “attribute-value” annotation that would make a querying feasible. Annotations that use “attribute value” pairs require users to be more principled in their annotation efforts. Users need to have good idea in using and applying the annotations or attributes. Even if the system allows users to annotate the data with such attribute-value pairs, the users are often unwilling to perform the task. Such difficulties results in very basic annotations that is often limited to simple keywords. Such simple annotations make the analysis and querying of the data cumbersome. Users are often limited to plain keyword searches, or have access to very basic annotation fields, such as “creation date” and “size of document” . In this paper, we propose CADS (Collaborative Adaptive Data Sharing) platform which is an “annotate-as-you-create”[1] infrastructure that facilitates fielded data annotation. A key contribution of our system is the direct use of the query workload to direct the annotation process, in addition to examining the content of the document. Our aim is to prioritize the annotation of documents towards generating attribute names and attribute values for attributes that will often used by querying users and these attribute values will provide best possible results to the user wherein users will have to deal only with relevant results.

Generation of sentences basically is the task of natural language generation [3], which requires knowledge about its syntax, semantics, ontology, morphology etc. Here, we have taken the task of sentence generation from a given input of bag of words. Our bag of words comprises of unordered words (which is basic unit of a sentence) taken from a grammatically correct sentence.

1.1 Contributions

- We present an adaptive technique for automatically generating data input forms, for annotating unstructured textual documents, such that the utilization of the inserted data is maximized, given the user information needs.
- We create principled probabilistic methods and algorithms to seamlessly integrate information from the query workload into the data annotation process, in order to generate metadata that are not just relevant to the annotated document, but also useful to the users querying the database.
- We present extensive experiments with real data and real users, showing that our system generates accurate suggestions that are significantly better than the suggestions from alternative approaches.
- The rule based technique can be used to generate sentence by leveraging the rules with the knowledge of syntax of a language or by exploiting the interdependencies between words in language.

2. Related Work

2.1 Review Stage

Calais: We use the Open Calais10 information extraction system, as a black box. Calais can recognize persons, locations, dates and other entities that are common in news articles. The entities extracted are fixed to a particular schema that we map to our own attributes. We annotate the documents and consider all the attributes that correspond to an entity. We use the Calais relevance score to rank the attributes. If the same attribute is annotated with multiple values we use the highest relevance score value to score it. If the Calais not yet working, we use NER model and

algorithms. Products have specialized attributes and hence we cannot use this generic extractor as a baseline [2], so we only use this strategy as a baseline for the Emergency data set.

In Figure 1(a) we show a report extracted from the National Hurricane Center repository, describing the status of a hurricane event in 2008. The report gives the current storm location, wind speed, warnings, category, advisory identifier number, and the date it was disclosed. Even though this is a text document, it contains implicitly many attribute names and values, e.g., (StormCategory, 3). If we had these values properly annotated (e.g., as in Figure 1(b)), we could improve the quality of searching through the database. For instance, Figure 1(c) shows three sample queries for which the report of Figure 1(a) is a good answer and the lack of the appropriate annotations makes it hard to retrieve it and rank it properly.

ZCZC MIATCPAT2 ALL
TTAA00 KNHC DDHHMM
BULLETIN
HURRICANE GUSTAV INTERMEDIATE ADVISORY NUMBER 31A
NWS TPC/NATIONAL HURRICANE CENTER MIAMI FL AL072008
600 AM CDT MON SEP 01 2008

EYE OF GUSTAV NEARING THE LOUISIANA COAST...HURRICANE FORCE WINDS OVER PORTIONS OF SOUTHEASTERN LOUISIANA... A HURRICANE WARNING REMAINS IN EFFECT FROM JUST EAST OF HIGH ISLAND TEXAS EASTWARD TO THE MISSISSIPPI-ALABAMA BORDER...INCLUDING THE CITY OF NEW ORLEANS AND LAKE PONTCHARTRAIN. PREPARATIONS TO PROTECT LIFE AND PROPERTY SHOULD HAVE BEEN COMPLETED. A TROPICAL STORM WARNING REMAINS IN EFFECT FROM EAST OF THE MISSISSIPPI-ALABAMA BORDER TO THE OCHLOCKONEE RIVER. GUSTAV IS MOVING TOWARD THE NORTHWEST NEAR 16 MPH...26 KM/HR... ON THE FORECAST TRACK...THE CENTER WILL CROSS THE LOUISIANA COAST BY MIDDAY TODAY. MAXIMUM SUSTAINED WINDS ARE NEAR 115 MPH...185 M/HR...WITH HIGHER GUSTS. GUSTAV IS A CATEGORY THREE HURRICANE ON THE SAFFIR-SIMPSON SCALE.

(a) Example of an unstructured document

Storm Name = 'Gustav'
Storm Category = 3
Warnings = 'tropical storm'

(b) Desirable annotations for the document above

Q1: Storm Name = 'Gustav' AND Warnings = 'flood'
Q2: Storm Name = 'Gustav' AND Storm Category > 2
Q3: Document Type = 'advisory' AND Location = 'Louisiana' AND Date FROM 08/31/2008 TO 09/30/2008

(c) Queries that can benefit from the annotations

Figure 1(a), (b), (c)

2.2 Information Extraction Algorithm

1) Select a text file
2) Parse the text file. Ignore stopwords from it and count frequency of high querying keywords which will be important for content based search.
3) Maintain frequency count of these keywords appearing in only single document.
4) Upload the file on to the server
5) Then fill all the annotations which are relevant to the document which can be useful for querybased searching.

Example : year = 2012, location = 'Nashik', author =' Bill Gates' etc.

2.3 N-gram Language Model

Considerable amount of work has been done in corpus based sentence generation. As mention in paper they have worked on sentence generation by using two lexical resources I) Case frame data which describes what kind of noun related to each predicate and II) Google N-gram frequency. Inputs to system were verb and noun. Case frame data used to select appropriate Japanese postposition for each noun and Google N-gram was used to check the correctness of a sentence by finding the co-occurrence of N-gram. In paper, the author has worked on String Regeneration task which is a task of text-to-text generation. Author has proposed an approach to recover original sentence from a bag of words. The words were taken from a fluent grammatical sentence. They used graph based approach to compute the highest probability permutation for a given input. They took the task of finding the permutation with highest probability for given input equal to finding the shortest tour in the graph. They used N-gram language model to the string regeneration task. In paper, author has worked on Statistical Sentence Generation by using the concept of Forest [5]. They applied the concept of packed set of tree to overcome the problem of lattice and to take the advantage of compactness. We have tried to generate all possible correct sentences if we are given a bag of words.

The N-gram language model provides the probability of next word P(W) by seeing the history. It takes N as a window size and the probability of Nth element evaluated on the basis of previous (N-1) elements. The basic equation to calculate the probability of occurrence of next word on a given context of (N-1) elements by equation. N-gram model concentrates on local dependencies and also encodes linguistic information like syntax and semantics etc. The N-gram model is local dependent because it takes the window size of N and predict only by taking the context of (N-1) words. We have used bigram model to generate the probability matrix from the text corpus. From the probability matrix we have generated permuted sentences from the input bag of words by applying DFS technique. Each sentence in the text corpus is wrapped within starting symbol (<s>) and ending symbol (</s>). However, N-gram model faces the problem of sparse data. The probability of infrequent bigrams is zero. To overcome this problem the Kneser-Ney smoothing Technique is applied to provide some probability mass to the zero probability N-grams.

2.4 Part of Speech Tagging

Part of speech tags represent what the role a word is playing in the sentence. To capture the syntax of language or to learn which word is preceding which another word we have used trigram model and have extracted the trigrams from the second annotated corpus of POS tags to generate a list of valid trigrams. We have applied Stanford Part-of- Speech Tagger to the first text corpus. In the process we have obtained the annotated corpus of POS tags. We have used

Part-of-Speech Trigram as a matching template instead of trigrams of words to find the correct sentence among generated candidate sequences.

2.5 Dataset Description

The textual information contain in the first corpus were short stories and daily conversation. N-grams statics for corpus Bigram Probability Matrix was generated from this text corpus of sentences for all the possible bigrams.

- Generate candidate sequences.
- Provide rank to the generated candidate sequences.

At the former stage bigram model is used to generate the possible candidate sequences by using the DFS (Depth First Technique) filtering technique while the valid trigram templates of POS tags were used to provide rank to the generated sentences at the later stage.

2.6 Algorithm to generate sentences from a given input bag of words

1) Input:- Bag of words (W1, W2……..Wn-1, Wn) where Wn is the nth words.
2) Generate Bigram Probability Matrix for input bag of words.
3) Construct lattice of bigrams.
4) Apply DFS search to generate only candidate sequences.
5) Do Part-of-Speech tagging in each generated candidate sequence.
6) Decompose POS sequence of each candidate sequences in to trigrams.
7) Count the number of valid trigrams of POS tag in each sequence by checking each trigram of POS tag with the valid trigrams templates extracted from the annotated corpus of Part-Of-Speeches.
8) If two or more sentences have same number of valid POS trigram templates then the score of sentence will be taken in to account.

3. Background Theory

3.1 Datasets

Documents: For our experiments use document collections:

- The Emergency corpus consists of 270 documents, generated by the Miami-Dade Emergency Management Office. The documents are advisory, progress and situation reports submitted by various county stakeholders during the five days before and after Hurricane Wilma, which hit Miami-Dade county in October 2005.
- The CNET corpus consists of 4,840 electronic product reviews obtained from CNET7. The dataset contains different kinds of products like cameras, video games, television, audio sets, and alarm clocks.

Collaborative Annotation: There are several system that favor the collaborative annotation of objects and use previous annotations or tags to annotate new objects. There have been a significant amount of work in predicting the tags for documents or other resources.

Information extraction (IE): Information extraction[6] is related to this effort, mainly in the context of value suggestion for the computed attributes. (for an overview of IE.) We can broadly separate the area into two main efforts: Closed IE and Open IE. Closed IE requires the user to define the schema, and then the system populates the tables with relations extracted from the text. Our work on attribute suggestion naturally complements closed IE, as we identify what attributes are likely to appear within a document. Once we have that information, we can then employ the IE system to extract the values for the attributes. Open IE is closer to the needs of CADS. In particular, Open IE generates RDF with no input from the user. Open IE leads to a very large number of triplets, which means that even after the successful extraction of the attribute values, we still have to deal with the problem of schema explosion that prevents the successful execution of structured queries that require knowledge of the attribute names and values that appear within a document. In principle, we could use Open IE, and then pay-as-you-go solutions for identifying equivalency relations across attribute names: however, it is much better to deal with the problem early-on, during document generation, instead of trying to fix issues that could be prevented with proper design. The CIMPLE project uses IE techniques to create and manage data-rich online communities, like the DBLife community. In contrast to CIMPLE, where data is extracted from existing sources and a domain expert must create a domain schema, CADS is a data sharing environment where users explicitly insert the data and the schema automatically evolves with time. Nevertheless, the IE and mass collaboration techniques of CIMPLE can help in creating adaptive insertion forms in CADS.

3.2 Depth First Search in Sentence Generation

We have tested our system on different number of input words to check how many numbers of sequences it is able to eliminate at the run time. To make a window to cover more candidate sequence we have taken the value of as 100 which reduce the MAX probability up to some extent.

We have experimented with total 50 numbers of bag of words. Each bag of words contain 3 words (excluding <s>, </s>). Out of 50 total number of bag of word, 25 number of bag of words contains 2 correct sentences as output and the remaining 25 bag of words contain only 1 correct sentence as output.

In our work, we have observed that by taking N-gram of POS tags increases the frequency of correct sentences. So the model discussed here is able to provide better result by using Part-of-Speech trigrams as matching templates and can cover more number of sentences to generate**.** By seeing a particular sequence of POS tags model can infer to other sentence of same pattern, to check whether it is correct or not. Filtering technique that is Depth First Search applied to eliminate sufficient number of unnecessary path that may not lead to a possible correct sequence.

4. Conclusion

We proposed adaptive techniques to suggest relevant attributes to annotate a document, while trying to satisfy the user querying needs. Our solution is based on a probabilistic framework that considers the evidence in the document content and the query workload. The task of sentence generation has its usage in text summarization, question answering system etc. The focus of our task is to generate all possible correct sentences from a given bag of words.

5. Acknowledgment

The Author would like to thank Meenu Varghese Assistant Professor, Department of Information Technology, Ilahia College of Engineering and Technology, Muvattupuzha for her moral and technical support.

References

[1] J. M. Ponte and W. B. Croft, “A language modeling approach to information retrieval,” in Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, ser. SIGIR ’98. New York, NY, USA: ACM, 1998, pp. 275 281
[2] J. Madhavan and et al., “Web-scale data integration: You can only afford to pay as you go,” in CIDR, 2007.
[3] M. Jayapandian and H. V. Jagadish, “Automated creation of a forms-based database query interface,” Proc. VLDB Endow., vol. 1, pp. 695–709, August 2008.
[4] N. Ito and M. Hagiwara, "Natural language generation using automatically constructed lexical resources," in The 2011 International Joint Conference on Neural Networks (IJCNN), pp. 980-987, Aug 2011.
[5] Irene Langkilde, “Forest-based statistical sentence generation”. Proceedings of the first conference on North American chapter of the Association for Computational Linguistics, Seattle, Washington p.170-177, April 29-May 04, 2000.
[6] A. Jain and P. G. Ipeirotis, “A quality-aware optimizer for information extraction,” ACM Transactions on Database Systems, 2009.

Author Profile

Jisha James received the Bachelor of Technology degree in Information Technology from Mahatma Gandhi University, Kerala. She is currently doing Master of Technology degree in Computer Science and Engineering with Specialization in Information Systems from Mahatma Gandhi University, Kerala.

Meenu Varghese She is currently assistant professor at ICET, Muvattupuzha, Mahatma Gandhi University, Kerala.