International Journal of Engineering & Technology, 7 (1.3) (2018) 33-36

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Ontology based search result optimisation using singular matrix

R. Divya [1]*, S. Angel Latha Mary [2]

[1]Asst. Prof./CSE, Karpagam College of Engineering, Coimbatore, Tamilnadu, India
[2]Professor/CSE, Karpagam College of Engineering, Coimbatore, Tamilnadu, India
**Corresponding author E-mail: todivyasmailbox@gmail.com*

Abstract

In recent era, today a many firms share their service/product descriptions. With that, many meaningful information in the textual form is hidden under the unstructured text. Algorithms like information extraction enable the identification of structured relations and they does not produce an optimal result and it is much costlier to operate with headlines of a text which has no examples of the target structured information. We propose a new approach which facilitates the formation of a structured metadata by recognizing documents which are likely to have some type and this information is to be subsequently used for both segregation and search process. Our approach is based on an idea that some attributes of a text will match with the query object which acts as identifier both for segregation as well as for storage and retrieval. Our implementation results show that our approach provides some superior results than existing approaches which rely only on query content or on textual information, to discover the key attributes.

Keywords: *Semantic Analysis, Segregation Index Creation And Recommender System.*

1. Introduction

Many existing organization share their descriptions about products and services. For illustration, Scientific networks, social networks or disaster management group share their information. Prevailing technologies like content management software (e.g.:-Microsoft Share point) allows users to share documents and tag them in a improvised manner [1]. Like that, Google Base allows its users to define objects for their use either by choosing from predefined or to define their own attributes. This process may facilitate subsequent information discovery. Many annotation systems provide a single way for annotation:"un typed" annotation. Consider that a user may annotate a weather report using a tag such as "Storm Category 1".

In general the most effective expressive annotation strategies use "attribute-value" pairs as they can contain more un typed information than un typed approaches. In such cases, the above information can be entered as (Storm Category,1).A most recent work in using the most expressive queries is the "pay-as –you-go" querying strategies in Data space in which users provide the data integration hints at the query time. In such hypothesis based system, the structured information is already present and the difficulty is with matching the source attributes with the query attributes.

Some systems which don't have any basic idea about "attribute-value" annotation that makes the "pay-as-you-go" querying feasible."Attribute- value" pair based annotation requires users to be more focused on their annotation efforts. In such case, the users should know about the underlying architecture and associated field types. They should also be aware of when to use these field types individually. With architecture which requires some hundred's of information to be filled, the process becomes much complicated and congested. It results in the ignorance of annotation capabilities that is left to be used by the users [1, 5]. Even if some systems allow the users to annotate a document in a random manner. This task requires some effort.

The user should have unclear usefulness for subsequent searches in the future-who is going to use an arbitrary, undefined in a common schema, attribute type for future searches. Even if the attribute fields are limited to a particular number, we can't predict like what fields among them will be utilized to search for searching at the future. Such issues results in naming a document with the very basic keywords. Such simple things make the analysis and querying of a data to be more tedious. In such case, users were often limited with plain text searches or to have access to very basic annotation fields such as "creation date", "owner of a document" and so on.

In this paper, we propose a cost segregation approach which is similar to CADS(collaborative Adaptive Data Sharing Platform).It is an approach which has two ways of segregating a document.Users if pose some query in a search box at normal search in windows, it checks only for the match of keyword with the title of the document. We are going to eliminate this and we improve it to search by its content too.

2. Related work

(i)Collaborative annotation: Systems like IBM MPEG-7 tool favor this type of annotation for an object and uses the previously used tags for annotations of new objects. An eloquent amount of work has been done to predict tags for documents or resources like web pages images, videos[13],[14],[15],[16],[17], From the users perspective and involvement, this approach takes different forms on what is anticipated as an input to the system. However the goals are similar to predict the missing tags that are related to an object. We feel that our approach is different ,as we consider only the basic keywords to be matched with the content of a document.

When compared with many other approaches, Clarity is a primary goal as we expect that the annotator may improve the annotations on process. But the discovered tags assist on quest of retrieval as an alternative to bookmarking.

*(ii)Data spaces and pay-as-you-go integration:*The consolidated model for CADS is quiet similar to a data space [18], in which a heterogeneous source is proposed for a loosely integrated model. A cardinal difference is that data space use blending of existing annotations to produce solutions for a query. But our work evince the appropriate annotations at the insertion time, by considering the query workload to identify the important attributes to add.

(iii)A real Time application-Google Base: A real time application which is a related data model is Google-Base [1],in which users can specify attribute/value pairs of their desire.

(iv)Information Extraction: Information extraction is mainly related with the context of value suggestion for the computed attributes. We can classify the IE into two namely open IE closed IE. Closed IE is much cumbersome but open IE is close to CADS approach .We use open IE.

3. Limitations in the existing work

Our inspiring frame work is a disaster management situation, inspired by the experience in building business continuity information network [3] for disaster situations in India. During calamities, we have many users and concerns proclaiming and investigating information. For example, in a hurricane situation, local government firms report shelter location, damages in structures, or structural warnings. Climatologically based Agencies report the status of the Hurricane, its position, and particular warnings. Enterprise owners describe the status and needs of their stores and personnel. Participants share their activities and look for critical needs. The information produced and investigated in this domain is dynamic and incalculable, and agencies have their own protocols and formats of sharing data. For example, The Miami-Dade county emergency office publishes hourly document reports. Further, learning the representation from previous calamities is hard, as new situations, needs and requirements arise. In fig.1a, we show a report extracted from the National Hurricane Center repository, narrating the status of a hurricane event in 2012.The report gives the storm location, wind speed, warnings, category, advisory identifier number, and the date it was revealed. Despite the fact, this is a text chronicle, it contains absolutely many attributes names and values, for example, (storm category, 1).If we had these values appropriately annotated (e.g., as in fig.1b), we could improve the standard of the penetrating through the database as well as the ability of a person to understand the structured information from an unstructured text or an occurrence, Fig.1c shows three specimen queries for which the report of Fig.1a is a good answer and the lack of the appropriate annotations makes it hard to retrieve it and rank it properly and is still in the "document generation "phase, despite the fact that the techniques can also be used for post generation document annotation. In our framework the originator generates a new document and uploads it to the warehouse. After the upload, CADS examines the text and originates an adaptive insertion form. The form holds the best attribute names chronicle text and information need (query workload), and the most feasible attribute values given the chronicle text. The originator can inspect the form, recast the generated metadata as necessary and store the annotated chronicle for storage.
We should note that the incorporating fielded metadata is not the only framework in which the CADS procedures are applicable.

With receding waters, Chennai city and suburbs on Saturday battled hard to pick up pieces of life but occasional heavy rains threatened to revive the ghost of flooding as lakhs of people in the worst-hit areas faced acute short supply of essentials including water, power, milk and food items.

Intermittent rains, occasionally heavy, in the city in areas like Kodambakkam, T. Nagar, Adyar and Kotturpuram and suburban Tambaram today threatened to revive the ghost of flooding again but the weatherman has forecast only light rains for Chennai in the next 24 hours. Heavy to very heavy rains have been forecast for south coastal and interior districts and Puducherry.

Arterial Mount Road and several other important roads were opened for traffic on Friday after three days of disruption bringing a slight sense of normalcy as water levels in Adyar and Cooum rivers and other channels came down following reduced discharge of water from Chembarambakkam, Puzhal and Poondi, and Red Hills rivers dotting the city's outskirts.

Fig a: Example of an unstructured document

Storm Name='OOKI FLOOD'
Storm Category=2
Warnings='FLOOD'

Fig b: Desirable annotations for the document above

Q1: Storm Name='OOKI FLOOD' AND Warnings ='flood'
Q2: Storm Name='OOKI FLOOD' AND Storm Category>2
Q3: Document Type='Advisory' AND Location='India' AND Date FROM 11/31/2017 TO 12/31/2017

Fig. C: Queries that can be benefitted from the annotation

Important metadata from the documents, so that this data can be used efficiently in the future (e.g., using a data spaces approach).If we use mechanized information extraction algorithms(IE) to squeeze targeted relations from the chronicle (e.g., addresses of evacuated buildings),it is important to process only chronicles that literally contain such data. When we process documents that do not hold the targeted data and we use automated information extraction algorithms to squeeze such fields, we often face a significant numbers of false positives, which can lead to noteworthy standard problems in the data[4].Likewise, if the chronicles are processed by humans(i.e., where there is low probability of false positives),requesting humans to inspect chronicles, where no relevant data is present, is valuable and counterproductive. For example, if only 1% of the chronicles contains data about the address of expelled buildings, it is going to be unnecessarily valuable to query humans to inspect all documents to identify such data. It is much better to target and process only optimistic chronicles, with high probability of containing relevant data.

Going back to a calamities management motivating framework, after the user submits the hurricane advisory document of fig.1a,CADS studies the content and finds that the following attributes types are relevant and present in the data:"storm name", "storm category", and "warnings".Fig.2 presents the adaptive insertion process for that chronicles. The system adds the proposed attributes to a set of default attributes like:"Document Type", "Date", and "Location", which are the basic metadata that the user always provides, as defined by a domain expert. This adaptive generation of metadata forms allows for much more efficient metadata generation.(of course, the user can also add new

attributes, which are not proposed by the adaptive form.)As we are going to see later, our CADS system prioritizes and propose first attribute types that are commonly used by the users who issue queries against the database.
Another part is that the user can perform a process of retrieving the documents he saved ,to get some structured information. If the number of documents to be investigated is less and if their content pages are less, a user may do that manually or may search them by using search that is what inbuilt in their system's operating system. A major problem that is existing with the operating system's inbuilt search mechanism in Windows 7/8/XP is that the concern what they give it to the data is much less.

So, If a person is searching for a document that has some words, if that word is existing in a document or a file, it retrieves them and displays them. Along with that it also displays other files like Images, Video clips, Movies and etc. which has the particular word. Note that the user posed a query only to retrieve the matching documents and not the other files that matched with the keyword.

In some cases, a user may know the meaning of a word which he searches. But , may not be sure of the word. In such cases, the user has to find the meaning of the word first and then has to search either manually or by using the inbuilt search in an operating system.

4. Proposed work

In this paper, We propose two things:

a)We present an adaptive technique for automatically generating segregated data from the whole unstructured documents by annotation such that the utilization of the data when a query is given is maximum by considering the basic keywords.

b)We present a different search strategy in which a particular keyword is given, the search engine maps the word to its relevant meaning. Then, the search process searches by following the pattern as below,

i)T[Best].
ii)T[Good].
iii)T[Empty].

We create this by examining principled probabilistic methods and algorithms to extract keywords from query workload and to use those keywords to segregate the document as T[Best] ,T[Good] or T[Empty].This separation makes the search process to be an easier one. This separation, helps us to display the documents with the exact word as similar to keyword given in T[Best] strategy and the documents with the meaning for a given word in T[Good].We propose this, with real datasets and real users to get the accurate results.

Example 2.

Attribute Name	Attribute Value
Storm Name	OOKI FLOOD
Storm Category	2
Warnings	FLOOD
Storm Speed	75 Kmph
Location	Chennai
Max Wind Speed	75Kmph

1.The query workload W contains a set of conjunctive queries of the form Q=(q1 ^,.., ^ qm), where each qi is a triplet (Aj, p,V), where Aj is an attribute value, p is a predicate (e.g.=, >,<), and V is an attribute value. The queries in the workload express the information need of the users and we expect similar queries to be asked in the future.
2. The answer to a query Q are all the documents in D, with annotations that satisfy the conditions of Q. For simplicity, and without loss of generality, we only consider the equality predicate in this work, although we also show some examples with more complex predicates (range condition in Fig. 1c).

Example
If a particular document is having information about "Apple company" the proposed work in this paper is as follows: Initially the engine gets the keyword for which a document is to be retrieved. Then the process checks the word in the T[Best] database. If that keyword is already present, it gives them. Else, it considers that as a new search and examines all the files and directories of the selected drive to get that document. Note , Here the search process searches the keyword in all the contents of a document and not with its saved name. If the file is not found it stores, the keyword in T[Empty].If the Document is found, it displays them. Then it maps the Keyword with its meaning. Then searches for the meaning as how it searched before. If the document is found it displays them in T[Good].

Example 3:

Attribute Name	Attribute Value
Storm Name	OOKI FLOOD

Attribute Suggestion
The problem that we are going to deal is attribute suggestion problem. To potentially solve this problem we identify and suggest attribute for a document d with two properties:

1) Querying value (QV)-It is the collection of basic keywords for a content type of a document.

2) Content value (CV)-It is the set of keywords from the content of a document (dt).

Based on the above two properties we process a document, we find the keywords in CV which matches with the QV. The real play starts here, when a keyword is read in a document it is processed with the QV to find the type of the CV. If exact match is found type of the document is thus identified. Then the document is annotated based on the type of the document that is identified by the result with the QV.

We first introduce the two optimal suggestion techniques namely,

OPT Full Match:
It is a technique which uses the subset of the ground truth attributes for each document that satisfies the maximum number of queries. It is an NP-hard problem. For simple workload it works well at the same time for a huge workload it take some significant amount of measurable time.

OPT Partial Match:
It is a technique which maximizes the number of query conditions satisfied. It is found by making a single pass on workload.

Keyword Fetching and Mapping

This module plays a dual part in which depending upon the user's choice the search and retrieve process initiates.

Search process-When the user gives a document it actually checks each and every word of the document and compares it with the pre-saved keywords in the database. If it matches with any of those words the meaning of the word is identified by mapping..

Recapture and Segregation-It is a step at which a user initiates to fetch some document. When a user gives a query keyword to search, it receives it, searches in the database for matching documents with the desired annotation as either T[Best],T[Good] or T[Empty] and displays it.

Searching a content: If a particular user gives some queries or keywords to search in order to display a document, it searches with the relevant type if it is identified or else as a whole it is processed.

5. Conclusion

We propose an adaptive technique to deal with relevant attributes to annotate a document by satisfying the users querying needs. Our solution for annotation-attribute suggestion problem is not based on the probabilistic model or prediction but it is based on the basic keywords that a user can use to query a database to retrieve a document. Experiment show that QV and CV approach is much useful in predicting a tag for a document and thus prediction is also based on QV and CV which greatly improves the utility of shared data.

References

[1] R.T. Clemen and R.L. Winkler, "Unanimity and Compromiseamong Probability Forecasters," Management Science, vol. 36,pp.767779,

[2] C.D. Manning, P. Raghavan, and H. Schu¨ tze, Introduction toInformation Retrieval, first ed. Cambridge Univ. Press,

[3] P.G. Ipeirotis, F. Provost, and J. Wang, "Quality Management onAmazon Mechanical Turk," Proc. ACM SIGKDD Workshop Human Computation (HCOMP '10), pp. 64-67,

[4] R. Fagin, A. Lotem, and M. Naor, "Optimal AggregationAlgorithms for Middleware," J. Computer Systems Sciences,vol. 66, pp. 614-656, http://portal.acm.org/citation. cfm?id=861182.861185, June 2003.

[5] K.C.-C. Chang and S.-w. Hwang, "Minimal Probing: Supporting Expensive Predicates for Top-K Queries," Proc. ACM SIGMOD Int'l Conf. Management Data, 2002.

[6] G. Tsoumakas and I. Vlahavas, "Random K-Labelsets: An Ensemble Method for Multilabel Classification," Proc. 18th European Conf. Machine Learning (ECML '07), pp. 406-417,

[7] M. Miah, G. Das, V. Hristidis, and H. Mannila, "Standing out in a Crowd: Selecting Attributes for Maximum Visibility," Proc. Int'lConf. Data Eng. (ICDE), 2008.

[8] P. Heymann, D. Ramage, and H. Garcia-Molina, "Social Tag Prediction," Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '08), pp. 531-538,

[9] Y. Song, Z. Zhuang, H. Li, Q. Zhao, J. Li, W.-C. Lee, and C.L. Giles,"Real-Time Automatic Tag Recommendation," Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '08), pp. 515-522, http://doi.acm.org/10.1145/1390334.1390423, 2008.

[10] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green, "Automatic Generation of Social Tags for Music Recommendation," Proc.Advances in Neural Information Processing Systems 20, 2008.

[11] B. Sigurbjo¨rnsson and R. van Zwol, "Flickr Tag Recommendation Based on Collective Knowledge," Proc. 17th Int'l Conf. World Wide Web (WWW '08), pp. 327-336, http://doi.acm.org/10.1145/1367497.1367542, 2008.