AUTOMATIC TAXONOMY CONSTRUCTION IN INSURANCE BASED SEARCH USING SVM CLASSIFICATION

Ms. A.CATHERIN SHARMILA
ME-II Year
Dept. of Computer Science & Engineering
Engineering College,
Trichy-9
cathrinsharmila@gmail.com

Mr. P. SUDHAKARAN
Associate Prof.& Head
Dept.of Information TechnologyOxford
Oxford Engineering College,
Trichy-9
karan_sudha@rediffmail.com

***Abstract*- Taxonomy is very essential for any information to be organized. Itis designed to improve relevance in vertical search and more useful for web search query. The most public method for constructing taxonomy is manual construction. Due to information growing, building taxonomy for such information manually was time consuming and maintenance was difficult. So this system presents taxonomy construction techniques using SVM classification available for easier construction of taxonomy or generating taxonomy automatically.Previously the system compared the KNN with∈NN.In proposed system compare the SVM with (∈NN) for assess the time complexity.**

***Index Terms*- Nearest Neighbor Approximation, Singular value Decomposition, Support Vector Machine.**

I. INTRODUCTION

Taxonomy is a process of classifying content & organizing. It is an organized set of words used for organizing information and intended for browsing. Building taxonomy [1] is an essential element of many applications. For example, in web search, organizing domain-specific queries into a hierarchy helps for better understand the query and improve search results or help with query refinement. Taxonomy comes from "Taxos" means ordering and "nomos" means rule. This has been widely used in websites for classification of web pages or resources. This taxonomy similar to ontology. The difference between them is taxonomy is usually only a hierarchy of concepts (i.e) the only relation between concepts is parents/child, subclass/superclass and ontology is arbitrary complex relations between concepts can be expressed too (X married to Y or A works for B).The taxonomy of entities is trees whose
nodes areconsidered with entities which expect to occur in web search query. These trees are used to match keywords from the search query with keywords from answer. It use information which acquired from a general-purpose knowledgebase and a search engine to enrich the given set of keyword phrases. The knowledgebase use database, which has been constructed for search the details. The core of database consists of a large set of is-a relationships, which are extracted from a text corpus of data. The knowledgebase facilitates deriving concepts from keyword phrases and use the concepts to enrich the keyword phrase. In order to model text for inference, need to make the knowledge in probabilistic. To this end, introduce a set of probabilistic measures.

A keyword phrase may be syntactically and semantically complicated and require meaningful terms. Moreover, each concept is related with a probability score that indicates the powerof the concept. Then use features that consist of concepts and contexts to denote the data. Since even a short piece of text may contain multiple topics or concepts, it is important to rank them by their significance. The most time consuming process in agglomerative clustering lies in searching all pairs of cluster to find the best pairs to merge. If reduce the search space for Bayesian Rose Trees by pruning the pairs of keyword clusters, then it can reduce the cost of searching agglomerative clustering. It recommends a set of nearest neighbors for each data point. Then the search complexity only depends on the number of the nearest neighbors. So previously implement KNN and ∈NN approximations to improve the search results and reduce the time complexity.

II. PROBLEM DESCRIPTION

The system presents an approach for building a domain specific taxonomy from a set of keyword phrases with knowledge and contexts. First, obtain knowledge and contexts related to the keywords. In approach, first obtain knowledge (concepts that correspond to each keyword phrase) using a technique called short text conceptualization and a general purpose knowledgebase, then obtain contexts by submitting the queries to a search engine to recover all snippet words in insurance applications. Second, build the taxonomy using a Bayesian rose

tree clustering approach [3] that can automatically derive a domain-dependent taxonomy from a set of keyword phrases by using both a general knowledgebase and context. First, deduce concepts with the technique of conceptualization and mine context information from a search engine, and then induce the new taxonomy using a Bayesian rose tree. Moreover, nearest-neighborhood-based methods to speed up the original Bayesian rose tree algorithm. Particularly, the algorithm reduces the time and memory cost significantly. It also conducted a set of experiments to demonstrate the effectiveness and efficiency of the algorithms. Figure 1 shows the pipeline of taxonomy construction.

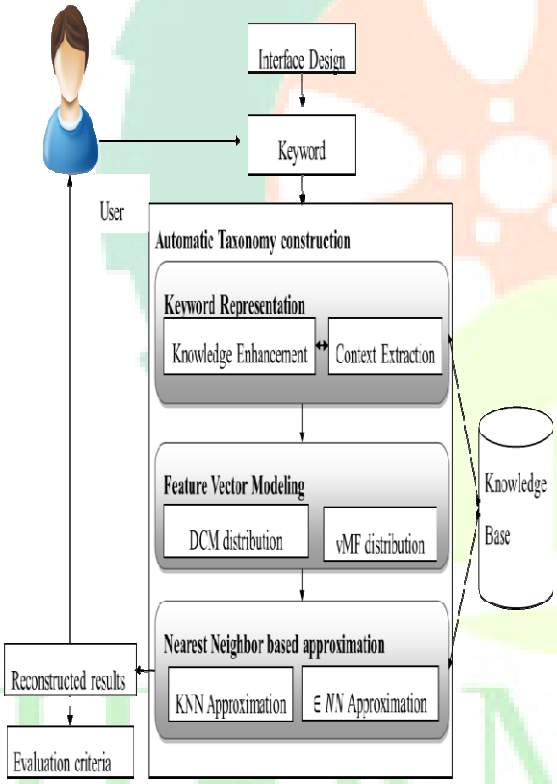

Fig. 1. Pipeline of taxonomy construction

A. Keywords Representation

In keyword representation, construct preference database that contains bag of words. The model simplifying representation used in NLP and information retrieval (IR). In this model, a text (such as a document) is represented as the bag (multi-set) of its words, disregarding grammar and even word order but keeping multiplicity. Building a domain specific taxonomy from a set of keyword phrases augmented with knowledge and contexts. And calculate P(instance|concept), P(concept|instances), P(attribute|concept) and P(concept|attribute) as

$$P(\text{instance}|\text{concept}) = \frac{n(\text{instances}, \text{concept})}{n(\text{concept})}$$

Where n(instance; concept) denotes the number of times instance and concept co-occur in the same sentence with the patterns preferred database used for extraction, and n(concept) is the frequency of the concept.

B. Feature vector Modeling

In feature vector modeling, use the Dirichlet compound multinomial (DCM) [7] distribution and the Von Mises-Fisher (vMF) distribution to model the data. DCM integrates the intermediate multinomial distribution, and represents either more general topics or multiple topics.

DCM can be defined as

$$f_{DCM}(D) = \int_{\theta} \prod_{i}^{n} p(x_i|\theta)p(\theta|\alpha)d\theta$$

$$= \prod_{i}^{n} \frac{m!}{\prod_{j}^{V} x_i^{(j)}!} \cdot \frac{\Delta(\alpha + \sum_i x_i)}{\Delta(\alpha)}$$

Using this marginal distribution $f_{DCM}(D)$, integrate the weights into the keyword feature vector.

vMF [2] is the distribution that characterizes this type of directional data. In this modeling, set both the prior and likelihood distribution to vMF and the marginal distribution have the following form:

$$f_{vMF}(D) = \int_{\mu} \prod_{i}^{n} p(x_i|\mu, k)p(\mu|\mu_0, K_0)d\mu$$

$$= \frac{cV(K_0)\prod_{i}^{n} cV(K)}{cV(||K\sum_i x_i + k_0\mu_0||}$$

Based on these calculations, feature vectors are constructed and are used for further searching process.

C. Nearest Neighbor based approximation

In nearest neighbor approximation, analyzed two foremost types of nearest neighbor approaches for efficient taxonomy construction: 1. k nearest-neighbor (kNN) and2. ∈-ball-nearest-neighbor (∈NN). kNN finds k nearest neighbors for each data and is not worried aboutquantity of the data. ∈NN uses a spherical ball to bind all the nearest neighbors within the ball.

kNN- approximation

Using the kNN approach, first find the k nearest neighbors for each data point, and then check

the probability of merging within the neighborhood set. To find k nearest neighbors of a data, keep a minheap of of size k to keep the data with largest similarities scores. When new data comes and the similarity score is larger than the top value, then replace the top index with the new data sample. Using random projection before partitioning the data space. When searching for the nearest neighbors in the sub trees with overlapping partitions, Spill tree searches one branch only.

∈ NN approximation

∈NN-approximation, for each data point, it keeps its nearest neighbors whose similarities with the data samples arebigger than a pre-defined threshold∈. Using prefix and suffix filtering, to filter out the data points that do not satisfy certain constraints.∈NNwill not return any nearest neighbors.

III. PROPOSED SYSTEM

Taxonomies arevital to forming knowledge and information for centuries. Now with the huge development of web technology, almost all the modern websites use taxonomies to improve user experience. Taxonomies play two essential roles in search engines. The first one is straightforward: page navigation. A webpage is associated with its relevant classes. Therefore, under each category in the taxonomy are the related web pages linked to it. Once a user navigates to a particular category, can browse those pages and delivery into the ones of interest.

The other one is not as explicit: taxonomies provide useful features for ranking in the retrieval process. Many machine learning algorithms can be applied to training data to generate a classifier. The system use Linear Support Vector Machine (SVM) due to its high accuracy and speed. However, it was trained with older click logs and only used unigrams as features. It is widely known that SVM usually outperforms KNN and ∈ NN. The purpose of the comparison KNN and ∈ NN with in this experiment is not to qualitatively compare SVM and KNN and ∈ NN but to quantitatively assess the impact of the extra features.

A.Objective

Main objective of proposed system is implementing Support vector classification algorithm to reduce the time complexity at the time of key word search.Provide privacy based personalization approach to improve search mechanism in insurance application

B. System Architecture

In this Architecture, design the interface as insurance applications. The Insurance Taxonomy that has several ideal terms and synonyms. It was designed and built to address both the policy writing and the premium investing sides of the insurance industry. The taxonomy covers different types of insurance, policy terms. Design the search box that can be used to extract keyword whether it has preference or not.

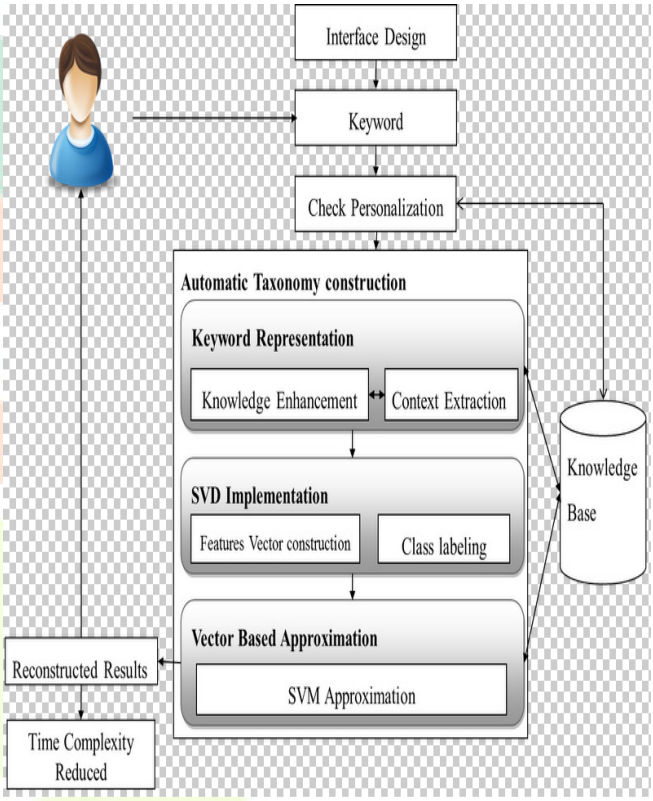

Fig. 2. System Architecture

C. Datasets Acquisition

In this module, upload the datasets like insurance application.Admin upload the datasets related to insurances.Datasets are in the form of documents.

D.Preprocessing

- Segregatethe each word in the documents.
- Eliminate noise words such as is, was, the and so on.
- Then using stemming words approach to remove prefix and suffix words.

E. Feature vector modeling

Feature extraction is a special form of dimensional reduction. Extract domain features through the evaluation of their weights in different

related domains. These weights are calculated using Singular value decomposition approach. It can construct m dimensional vectors with columns and n dimensional vectors with rows. These are constructed using terms which are not eliminated by preprocessing approach. Then calculate the singular values based term frequency and construct matrix as vectors. Finally predict the similarity values to label the data based on their semantics.

Term Similarity

In this module, implement Singular value decomposition techniques.Calculate term vectors based on term frequency. Term frequencies are constructed as term similarity matrix. The SVD theorem states:

Suppose **M** is a $m \times n$ matrix whose entries come from the field K, which is either the field of real numbers or the field of complex numbers. Then there exists a factorization, called a singular value decomposition of **M**, of the form

$$\mathbf{M} = \mathbf{U}\,\Sigma\mathbf{V}^*$$

Where

- **U** is a $m \times m$, unitary matrix,
- **Σ** is a $m \times n$ diagonal matrix with non-negative real numbers on the diagonal, and
- $\mathbf{V}^*$ is a $n \times n$, unitary matrix over K. (If $K = \mathbf{R}$, unitary matrices are orthogonal matrices.) $\mathbf{V}^*$ is the conjugate transpose of the $n \times n$ unitary matrix, **V**.

F. Vector based approximation

In vector based approximation, feature vectors mapped with the help of kernel function in the feature space. And finally division is computed in the feature space to separate out the classes for training data. The SVMs algorithm separates the classes of input patterns with the maximal margin hyper plane. This hyper plane is constructed as:

$$f(x) = \langle w, x \rangle + b$$

Where x is the feature vector, w is the vector that is perpendicular to the hyper plane and $b\|w\|^{-1}$ specifies the offset from the beginning of the coordinate system. Based on f(x) values, results are extracted and viewed by users.

Support Vector Machine

"Support Vector Machine" (SVM), a method for classification ofboth linear and non-linear data. In uses a nonlinear mapping to transform the original training data into a higher dimension. Within new dimension, it can searches for the linear optimal separating hyper plane. With an nonlinear mapping to a sufficiently high dimension data from two classes can always be separated by a hyper plane. Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine finds hyper plane using support vectors and margins which best segregates the two classes (hyper-plane/ line). Fig. 3 shows support vector machine classification.

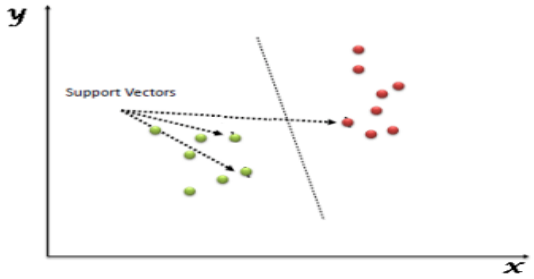

Fig. 3. Support Vector Machine

G. Reconstructed results

In this module can get the user preferred search results using preferred database. And then calculate the time complexity of the project based vector based approximations. SVM time complexity is reduced compared to existing nearest neighbor approximations.

IV. CONCLUSION

Emerging trends and products pose a challenge to modern search engines since must adapt to the constantly changing needs and interests of users. The main objective is study the problem of how to expand an existing category hierarchy for a search/navigation system to accommodate the information needs of users more comprehensively.In proposed system the system addresses the time complexity at the time of keyword search and personalized based checking using SVM Classification. This technique has high accuracy and speed. Term frequency is constructs as term similarity matrix using SVD.The purpose of the comparison (∈NN) and SVM to quantitatively assess the impact of the extra features.

V. REFERENCES

[1] Adams R. P., Ghahramani Z., and Jordan M. I. (2010), 'Tree-structured stick breaking for hierarchical data', In NIPS.

[2] Banerjee A., Dhillon I. S., Ghosh J. and Sra S. (2005), 'Clustering on the unit hyper

sphere using von mises-fisher distributions', Journal on Machine Learning Research, 6:1345-1382.

[3] Blundell C., The Y. W. and Heller K. A. (2010), 'Bayesian rose trees', In UAI, pages 65-72.

[4] Carlson A., Betteridge J., Kisiel B., E. R. H jr. and Mitchell T. M. (2010), 'Toward an architecture for never-ending language learning', In AAAI, pages 1306-1313.

[5] Chen W. Y., Song Y., Bai H., Lin C. J. and Chang E. Y. (2011), 'Parallel spectral clustering in distributed systems', IEEE Trans. Pattern Anal. Mach. Intell., 33(3):568-586.

[6] Dean J. and Ghemawat S., Mapreduce. (2004), 'Simplified data processing on large clusters', In OSDI.

[7] Elkan C. (2006), 'Clustering documents with an exponential-family approximation of the dirichlet compound multinomial distribution', In ICML, pages 289-296.

[8] Etzioni O., Cafarella M. and Downey D. (2004), 'Web scale information extraction in knowitall (preliminary results)', In WWW.

[9] Faure D. and N'edellec C. (1998), 'A corpus-based conceptual clustering method for verb frames and ontology acquisition', In LREC workshop on adapting lexical and corpus resources to sublanguages and applications, pages 5-12.

[10] Faure D. and N'edellec C. (1999), 'Knowledge acquisition of predicate argument structures from technical texts using machine learning', The system ASIUM. In EKAW, pages 329-334.