Constructing an Enriched Domain Taxonomy for Hindi using Word Embeddings

Vaishakh Keshava*, Pravalika Avvaru†, Sowmya Kamath S‡ and Geetha V‡
*Intuit Inc., Bengaluru, India
kvaishakhnambiar@gmail.com
†Language Technologies Institute, Carnegie Mellon University, Pittsburgh, USA
avvarupravalika@gmail.com
‡Department of Information Technology, National Institute of Technology Karnataka, Surathkal, India
{sowmyakamath, geethav}@nitk.edu.in

***Abstract*—Domain-specific taxonomies constitute a valuable resource as they offer extensive support in information retrieval related activities like browsing, searching, recommendations and personalization. Such taxonomies can bridge the gap between the lack of domain-specific querying knowledge in potential users and the actual content. In case of multilingual content, taxonomies can play a pivotal role in boosting search performance for content across language barriers. In this paper, a domain-agnostic framework for building an evolving, domain-specific taxonomy for the Hindi, given a set of well-organized data points is proposed. The approach is intended for designing a hierarchical taxonomy enriched with synonyms and other morphological variants using WordNet and Word2vec models respectively. The hierarchical structure acts as a base which binds the taxonomy to a given domain. Such enrichment can improve taxonomy coverage within the given domain. The focus is also on building a taxonomy that can self-evolve over time, with high precision and recall, with minimal manual effort.**

***Keywords*-Asian language processing; taxonomies; semantic processing; natural language processing**

I. Introduction

In a digital world, the ever-increasing volume of multimodal content in the form of textual and multimedia data is a significant challenge faced by applications that aim to facilitate intelligent access to it. Such information needs to be organized, classified and maintained based on their semantic content, for use in systems like search engines, recommendation systems, content management systems etc. Due to the massive amounts of available data, its effective management is often cumbersome and is beyond the scope of human capabilities. Context-aware applications often rely on and incorporate domain-specific knowledge for enhancing and streamlining the management of such content. Ontologies and domain-specific taxonomies can help in building efficient and accurate models, which are then helpful in classification, retrieval, recommendations etc. If the taxonomy is built on domain-specific knowledge, then it can help achieve high specificity and improved user satisfaction.

Taxonomy creation has been an area of active research interest and has been studied extensively by researchers. Some existing approaches focus on deriving domain-specific terms for taxonomy construction from the corpus itself [1], while others have incorporated Wikipedia for inferring relevant terms [2][3][4]. Creating taxonomies by extending the initial set of terms with Wikipedia and lexical resources like Wordnet [6][7][5] also have been proposed. In early corpus-based methods external knowledge was rarely used. The main aim was to extract taxonomic terms and hierarchical relations that capture the intrinsic characteristics of a given corpus. Term distribution statistics [8] or lexico-syntactic patterns [1][9] have been employed to this end. More recently, researchers have used concept taxonomy based techniques for novel applications like online product marketing analysis [12], social network mining [13][14][15], scientific big data analytics [16][17] etc.

The motivation for building a domain-specific taxonomy for India's official language, Hindi stems from the fact that more than 41.1% of the population are Hindi speakers. Like most other major languages of the world, Hindi too has a lot of dialects (like Haryanvi, Bundeli, Awadhi etc) and has around 295 million native speakers. Moreover, these speakers may not necessarily have expert language skills and domain knowledge. Already, the number of documents written in Hindi is also on a significant rise, due to the government's push towards active usage of the official language in administration and other affairs. Thus, the problem of searching across languages with different nuances and linguistic constructs is highly challenging. India being a multilingual country, this is a significant issue which requires dedicated effort. In this work, we focus on building a taxonomy for Hindi, stressing on domain-specificity and automatic evolution via self-learning. The framework comprises of modules for building and maintaining an evolving, domain-specific taxonomy, given an initial set of well-organized data points. The focus of this taxonomy is to bridge the gap between the limited set of terms used by people during search and the actual content itself, for improved ease of use.

The remainder of the paper is organized as follows. In Section II, we describe the proposed taxonomy structure in detail. Section III discusses the processes of taxonomy modeling and taxonomy population. Section IV presents the preliminary evaluation and results obtained using various evaluation metrics, followed by concluding remarks and directions for future work.

II. Proposed Taxonomy Structure

While designing the proposed domain-specific taxonomy, several factors have to be considered, to ensure that

978-1-5386-1981-0/17/$31.00 ©2017 IEEE

it can be useful and also self-evolving. We defined the following set of requirements for this purpose:

- *Ambiguity resolution:* As a word may mean different things in different domains and contexts, it is important that the taxonomy is capable of resolving this.
- *Automatic resolution of vocabulary issues*: Since Hindi has several dialects, the system should be capable of supporting them.
- *Dynamic*: The taxonomy should not be dependent completely on a particular static resource. It should be able to grow and evolve with time.
- *Relevance feedback from users*: Every system needs a provision to correct any errors and make necessary improvements. User feedback can help in incorporating this, thus ensuring continuous evolution.
- *Ease in maintenance*: There needs to be easy provision for incorporating changes to the taxonomy.

Taking the above defined criteria in mind, we propose a hierarchical taxonomy structure for Hindi. Figure 1 describes the proposed taxonomy structure in consideration to an example of fruit names used in Hindi. The nodes in the taxonomy are classified into a class node, instance node and reference node. There exist weighted links between these nodes and across the levels. This weight is a measure of relatedness or similarity between the nodes. Also, there can also be multiple nodes having the same name. As the taxonomy is not dependent on any static resources, it is flexible to changes. Node addition, removal or any modifications of the taxonomy structure is hence easily achievable, as the links between nodes do not have a mesh architecture. Further, the taxonomy can be augmented by incorporating user feedback.

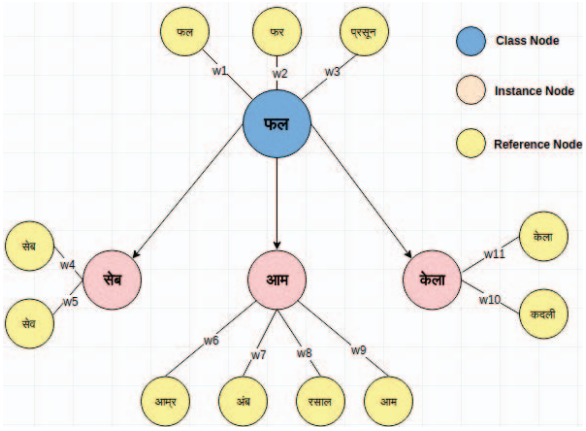

Figure 1. Proposed Taxonomy Structure

A. Taxonomy Node types

1) *Class Node:* A class node represents a conceptual entity which can be further classified into subclasses. A concept can also have instances. Say for example, if *fruit* is a conceptual entity, then *citrus fruits*, *pome fruits* are sub classes of it. At the same time *banana* is an instance of *fruits*. *Orange* will be an instance of *citrus fruits* which is a sub class of *fruits*. Similarly, a class too can have references. Fruits may be referenced with different terms at different places, which form its reference nodes. For example, a fruit is commonly called *phal* in Hindi. Other words like *phar* and *prasUn* also mean *fruit*, which are represented as *phal's* reference nodes in the proposed taxonomy.
2) *Instance Node:* An instance node is a leaf node in the proposed taxonomy and hence cannot be further classified further. For example, in the previous example, a *banana* is an instance of class *fruits*.
3) *Reference Node:* Reference nodes indicate the usage of a particular concept or its instance. A reference node is associated to a concept or an instance with an associated weight which signifies the relevance or importance of that node to its parent. For example, the word *Banana* (*kelA* in Hindi) has a synonym *kadalI*. So the associated weight between the two nodes will have to be high in the proposed taxonomy to capture their similarity.

B. Taxonomy Link Types

To indicate the relationships between taxonomy nodes, links are used. To indicate the importance of each link, associated weights are used. The various types of links and their associated weights are as below.

1) *Class Node to Class Node*: This is an unweighted link because a new sub class is added to the parent class when the dissimilarity between instances is higher than a particular threshold, and hence it is not required to represent one concept as partly similar to a sub concept.
2) *Class Node to Instance Node*: This is again modeled as an unweighted link as its not required to represent a relatedness measure between a conceptual entity and an instance of that entity.
3) *Class Node to Reference Node*: This is a weighted link as this weight denoted the relatedness measure between the class node and its reference nodes.
4) *Instance Node to Reference Node* : This is also a weighted link owing to the similar reason that it measures the relatedness/similarity between the instance node and its reference nodes.

C. Taxonomy Node Attributes

Each node in the taxonomy has a pre-defined set of attributes. Each node has a unique identifier called *Node ID*, *Parent ID*, *category*, *language* and *weight*. The details of these attributes are given in Table I.

III. Taxonomy Modeling and Population

As stated earlier, the two main issues that contribute to the widening gap between query terms and actual content are - the huge vocabulary needed for Indian languages and the lack of specific domain knowledge in users. So, it becomes essential to tackle these problems specifically in our proposed taxonomy design approach. With the development of the IndoWordNet [18] system,

Table I
ATTRIBUTES OF A NODE IN THE TAXONOMY

Attribute	Description
Node ID	Unique identifier for each node
Node Name	The name (text) of the node
Parent ID	Node ID of its parent node
Category	Whether node is a class/instance/reference node
Language	Language used
Weight	Weight with which the node is associated

Table II
WORD2VEC TRAINING PARAMETERS

Parameter	Value	Description
Size	100	dimensionality of feature vectors
Min count	5	words with total frequency lower than this are ignored
Window	10	maximum distance between current and predicted word
Algorithm	Skip gram	captures rare words/phrases.

the problem of the extensive vocabulary requirement has been mitigated to a certain extent. IndoWordNet is a linked lexical knowledge base which comprises of Wordnets of Indian languages like Hindi, Sanskrit, Bangla, Kannada etc. We used IndoWordNet to solve any cross-dialect and cross-language vocabulary issues, that are common in most Indian languages. IndoWordNet has a good collection of synsets, and the Hindi WordNet itself has over 39000 entries. For modeling the hierarchy, a multilingual agricultural thesaurus called AGROVOC [19] was used. AGROVOC[1] is a controlled vocabulary that provides over 32,000 concepts in 23 different languages for classes relating to agriculture, like, food, nutrition, agriculture, fisheries etc and is used extensively for indexing, retrieving and organizing data in agricultural domain.

To address the second problem, i.e. users' limited domain-specific knowledge, we designed techniques for incorporating all sets of related words for a given query word. Earlier works used thesauri like WordNets and external sources like Wikipedia for this purpose. We propose the use of the Word2Vec model [11], which is a word embedding based model that produces estimations of word representations in vector space. It uses shallow, two-layer neural networks which are trained to reconstruct linguistic contexts of words. Word2Vec takes a large corpus of text as input and this produces a vector space of several hundred dimensions. Each unique word in the corpus is represented as a vector such that words which have similar contextual representation in the corpus are located in close proximity to one another in the vector space. Mikolov et al [11] presented a comparative study of different algorithms and a discussion on the performance of Word2vec when compared to other semantic techniques like LSA. As Word2vec preserves the linear regularities among words, we adopted it for Hindi language modeling and representation. In our work, a Hindi language Wikipedia text dump was used[2] for training the Word2Vec model. The training parameters used are given in Table II.

The hierarchical skeleton of the proposed taxonomy is obtained from AGROVOC. From AGROVOC, we identify the nodes and classify them as class and instance nodes, which bind the taxonomy skeleton to the domain. The reference nodes are populated using Hindi WordNet and the Word2vec model trained on the Hindi language corpus.

[1]Available online at: http://aims.fao.org/vest-registry/vocabularies/agrovoc-multilingual-agricultural-thesaurus

[2]Available at https://sites.google.com/site/rmyeid/projects/polyglot#TOC-Download-Wikipedia-Text-Dumps

The synonyms of the nodes in the hierarchical taxonomy skeleton are obtained from WordNet. These are basically reference nodes and they are given a weight of 0.95. The taxonomy is further enriched with related words using the trained Word2vec model, which gives the set of related words along with a relatedness measure for each node in the taxonomy skeleton. A relatedness threshold of 0.5 was considered and only those words/phrases above this threshold were chosen. This relatedness score also is used as the link weight in the taxonomy. Finally, the taxonomy enriched with synonyms and related terms is stored as a model of domain-specific knowledge.

As the enriched taxonomy is hierarchical in nature with respect to the domain, it helps in ambiguity resolution in a straight-forward manner. The reference nodes cater to the problem of large vocabulary. In addition to this, the language attribute helps in identifying the dialect in which that particular terminology is used. It is very easy to add, remove or modify nodes in the proposed taxonomy structure, thus making it dynamic. Since there exist weighted links to reference nodes, it is possible to modify these weights based on the relevance feedback given by users for improving results over time.

IV. EXPERIMENTAL EVALUATION

To evaluate the suitability of the proposed taxonomy construction process and the effectiveness of the constructed taxonomy in applications such as context-aware search, we instigated a group of native Hindi speakers to provide a set of terms used in agricultural domain Some sample terms that occur in the set are *aam* (mango), *pIdakanAshi* (pesticide), *hal* (plough) etc. The set also included terms that were *perceived* to be agricultural terms by the native Hindi speakers. (for example, terms like *pashudhan* (a reference to livestock), *jutAyI* (a reference to ploughing), *mahAkshIr* (referring to sugarcane) etc belong to this category). Using this, we performed a lookup for the terms in the set in AGROVOC, WordNet and our proposed taxonomy. Table III tabulates the statistics on the terms discovered after this lookup process. It can be seen that, the proposed taxonomy successfully captured more than 87% of the terms put together by native speakers, due to its hierarchical structure, which is a 77% improvement over those discovered using AGROVOC and 35.9% increase over that of WordNet.

To further evaluate the constructed taxonomy, we also put together documents on the agricultural domain (collected from the Web using Google Search) and a collection

Table III
TERMS DISCOVERED USING THE LOOKUP PROCESS

Number of words/ phrases looked up	Number of words/phrases discovered		
	AGROVOC	WordNet	Our Taxonomy
100	49	64	87

of agriculture related short queries. The collected documents are fed into Apache Solr, which is a open source search platform built on Apache Lucene search library. To evaluate the suitability of the proposed taxonomy for IR, we used Solr to develop a search functionality with support for full-text indexing and search. The agricultural domain documents collection is fed into Solr and a full-text index is created. The set of queries are expanded using the taxonomy and then submitted to the search utility built on Solr for conducting a full-text search. Table IV depicts the performance during search with respect to query expansion using WordNet and then with the constructed taxonomy. The system was evaluated using a query set consisting of 10 agriculture-related search queries. The WordNet based search achieved a mean precision of 0.747 and mean recall of 0.722. The proposed taxonomy based search achieved much better results, with a mean precision of 0.765 and mean recall of 0.95. We observed a steep increase in the recall as the proposed approach also captures documents with synonyms and other related terms, due to the taxonomy integration. This gives f-measure value of 0.734 for WordNet integrated search as opposed to that of 0.847 for the proposed taxonomy integrated search. This clearly attests to the effectiveness and suitability of the proposed taxonomy based search methodology.

Table IV
PRECISION-RECALL PERFORMANCE FOR A PRE-DEFINED QUERY SET

Query	WordNet based Search		Taxonomy based Search	
	Precision	Recall	Precision	Recall
Q1	0.72	0.8	0.91	1
Q2	0.71	0.5	0.91	1
Q3	0.45	0.71	0.59	0.93
Q4	0.8	0.86	0.59	0.93
Q5	0.6	0.92	0.55	0.92
Q6	0.83	0.77	0.55	0.92
Q7	0.78	0.54	0.55	0.92
Q8	1	1	1	1
Q9	0.78	0.65	1	0.94
Q10	0.8	0.47	1	0.94
Average	**0.747**	**0.722**	**0.765**	**0.95**

V. CONCLUDING REMARKS

In this paper, a methodology for constructing a self-evolving, domain-specific taxonomy for the Hindi language, using the concept of word embeddings is presented. The hierarchical structure of the taxonomy and the weighted links between reference nodes help in capturing concepts relevant to a query even when users lacking in domain knowledge use the system. The design also incorporates relevance feedback for ensuring that the taxonomy continuously evolves with usage over time. As part of future work, we wish to further enrich the taxonomy and also explore other domains for which domain-specific taxonomies need to be constructed. We also plan to extend the concepts developed specifically for the Hindi language to other Indian languages to enable context-aware, cross-lingual, domain-specific search.

REFERENCES

[1] Caraballo, S., "Automatic construction of a hypernym-labeled noun hierarchy from text", 37th Annual Meeting of the ACL, pp. 120-126, ACL 1999

[2] Ponzetto, S., Strube, M, "Deriving a large scale taxonomy from wikipedia", 22nd National Conference on Artificial Intelligence, pp. 1440-1445, AAAI Press 2007

[3] Wang, Pu, and Carlotta Domeniconi, "Building semantic kernels for text classification using wikipedia," 14th ACM SIGKDD Intl. Conf. on Knowledge discovery and data mining, ACM, 2008.

[4] Hu, Xiaohua, et al, "Exploiting Wikipedia as external knowledge for document clustering," 15th ACM SIGKDD Intl. Conf. on Knowledge discovery and Data Mining, 2009.

[5] Snow, R., Jurafsky, D., Ng, A., "Semantic taxonomy induction from heterogenous evidence", 21st Intl. Conf. on Computational Linguistics, pp. 801-808, ACL 2006

[6] Stoica, E., Hearst, M.A., "Automating creation of hierarchical faceted metadata structures", HLT/NAACL Conference, 2007

[7] Dakka, W., Ipeirotis, "Automatic extraction of useful facet hierarchies from text databases", 24th IEEE Intl. Conf. on Data Engineering, pp. 466-475, IEEE 2008

[8] Sanderson, M., Croft, B., "Deriving concept hierarchies from text", 22nd Annual Intl. Conf. on R&D in Information Retrieval, pp. 206-213, ACM 1999

[9] Hearst, M., "Automatic acquisition of hyponyms from large text corpora", 14th Conference on Computational Linguistics, pp. 539-545, ACL 1992

[10] Cimiano, Hotho, and Staab, "Learning concept hierarchies from text corpora using formal concept analysis.", J. Artif. Intell. Res.(JAIR) 24.1 (2005): 305-339.

[11] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. and Dean, J., "Distributed representations of words and phrases and their compositionality", In Advances in neural information processing systems (pp. 3111-3119).

[12] Kaushik, R., Chandra, A., Mallya, D., Chaitanya and Kamath, Sowmya S, "Ontology based Approach for Event Detection in Twitter datastreams", In Region 10 Symposium (TENSYMP), 2015 IEEE (pp. 74-77). IEEE.

[13] Liu, J., Shang, J. and Han, J., "Phrase Mining from Massive Text and Its Applications", Synthesis Lectures on Data Mining and Knowledge Discovery, 9(1), pp.1-89.

[14] Lau RY, Li C, Liao SS, "Social analytics: learning fuzzy product ontologies for aspect-oriented sentiment analysis", Decision Support Systems, 2014; 65:80-94.

[15] Ali F, Kwak KS, Kim YG, "Opinion mining based on fuzzy domain ontology and Support Vector Machine: A proposal to automate online review classification", Applied Soft Computing, 2016;47:235-250.

[16] Gordon J, Zhu L, Galstyan A, Natarajan P, Burns G, "Modeling Concept Dependencies in a Scientific Corpus", In ACL (1) 2016.

[17] Osborne F, Motta E, "Mining semantic relations between research areas", The Semantic WebISWC 2012, pp 410-426.

[18] Bhattacharyya, P., "IndoWordNet", In The WordNet in Indian Languages (pp. 1-18). Springer Singapore, 2017.

[19] Caracciolo, C., Stellato, A., Morshed, A., Johannsen, G., Rajbhandari, S., Jaques, Y. and Keizer, J., "The AGROVOC linked dataset", Semantic Web, 4(3), 2013, pp.341-348.