Latent Dirichlet Allocation based Semantic Clustering of Heterogeneous Deep Web Sources

Ali Daud, Umara Noor, Ayesha Manzoor
International Islamic University, Islamabad
Pakistan
ali.daud@iiu.edu.pk, umara.zahid@iiu.edu.pk, ayeshamanzoor@ymail.com

***Abstract*:** *Over the years a critical increase in the mass of the web has been observed. Among that a large part comprises of online subject-specific databases, hidden behind query interface forms known as deep web. Existing search engines are unable to completely index this highly relevant information due to its large volume. To access deep web content, the research community has proposed to organize it using machine learning techniques. Clustering is one of the key solutions to organize the deep web databases. Existing clustering methods do not encounter semantic relevance among deep web forms. In this paper, we propose a novel method DWSemClust to cluster deep web databases based on the semantic relevance found among deep web forms by employing a generative probabilistic model Latent Dirichlet Allocation (LDA) for modeling content representative of deep web databases. A document comprises of multiple topics, the task of LDA is to cluster words present in the document into "topics". The purpose of the parameter estimation process in the underlying model is to discover the document's topic and tell about its proportionate distribution in documents. Deep web has a sparse topic distribution. Due to this reason we have proposed to use LDA that is supposed to be a good clustering solution for the sparse distribution of topics. Further we employ a rich set of metadata as our content representative that comprises of form contents (single attribute/ multiple attributes) and page contents. Experimental results show that our proposed method clearly outperforms the existing non-semantics based clustering methods.*

Keywords: Latent Dirichlet Allocation, Deep Web Mining, Soft Clustering, Topic Models, Semantics

Received: 2 August 2014, Revised 10 September 2014, Accepted 21 September 2014

1. Introduction

Since the last two decades, the World Wide Web has become an ultimate information repository that is widely used for searching and publishing information. Further the pervasiveness of internet has made the web a preferred medium for information transfer and commerce among leading businesses. Therefore researchers have been focusing on various web analysis and mining techniques in order to discover useful patterns from various perspectives that will facilitate in a more intelligent use of the web. Research in web mining can be classified into three areas i.e. web usage mining, web structure mining and web content mining. Web usage mining involves statistical methods to discover web usage patterns among various classes of users. Web structure mining involves hyperlink structure analysis and document structure analysis techniques that are usually based in graph theory. Web content mining involves classification, extraction and integration of web content. Web mining research adopts techniques from other research areas such as database, information retrieval,

artificial intelligence and natural language processing [1].

Web content can be classified into two broad categories: surface web and deep web. The surface web comprises of static content visible to web crawlers for indexing. The deep web comprises of databases hidden behind HTML forms. Since the advent of database technology to the web, the development of dynamic web sites with back end database have critically increased the mass of whole web. This information is not directly accessible to automated systems like crawlers; rather they are retrieved by online querying the database servers.

The size of the deep web content is much greater than surface web. So far an exact yearly growth rate in the number of deep web sites and its mass has not been determined by the research community. However there are a few estimates outlined in a few studies to depict massive increase in size of deep web. It was estimated in a survey performed in 2000, that there are 43,000-96,000 deep web sites having a mass of 7,500 terabytes [2] which is 400 times greater than the surface web. A more recent survey performed in 2007 give us with an estimated value of 25 million deep web sites [3]. What makes deep web so significant is the demand and relevance of its content regarding web search rather than its mass. In a survey, it was observed that deep web contents are at least 1,000 to 2,000 times more relevant to user information needs as compared to surface web [4].

Existing research interests in deep web take account of finding efficient techniques for 1) locating deep web entry points 2) crawling deep web content 3) organizing deep web content through different approaches of clustering and classification 4) integrating information from multiple deep web sources and 5) deep web source ranking. Here in this research work we focus on clustering of deep web sources. The reason for adopting a clustering approach instead of classification is that the deep web comprises of a large number of domains. A single deep web source may belong to multiple domains. Using a supervised learning approach like classification intends to restrict the number of domains and also involves labeling cost. In order to observe different structures of deep web we consider clustering to be a more appropriate solution for organizing deep web sources than classification. The ultimate goal of clustering deep web sources is to make them accessible on user's request as compared to surface web. Also user's satisfaction over the retrieved content is improved with short navigation pathways. Further it can help improve information integration by eliminating relevance determination problem.

Several deep web clustering approaches have been proposed [5-8]. Existing clustering works cluster sources based on the visible textual features of the deep web interface form. Therefore they employ all the rudiments of document clustering i.e. using an appropriate approach for document modeling (vector space model, bag of words model). So far no such approach has been proposed that encounter semantics of the words found on the deep web form interface in order to cluster sources. In this paper, we have proposed a novel approach for clustering deep web sources i.e. Deep Web Semantic Clustering (DWSemClust) which is based on latent dirichlet allocation [9]. LDA is a graphical topic model that clusters deep web sources based on topics. Each deep web interface is considered a single document that is a mixture of a number of topics. Based on the words found, each word's formation is assigned to one of the document's topics. Our approach shows better performance for improved deep web clustering in comparison to non-semantics based existing methods. The contribution of this work is usage of LDA based semantics for mining heterogeneous deep web sources and experimental verification of the effectiveness of the approach.

The paper is organized as follows. Section 2 highlights the limitations in existing works which ultimately became the motivation to conduct this research activity. Section 3 discusses the related work. Section 4 provides the details of our proposed approach. In section 5 we formally formulate our research problem. Section 6 provides experimental setup, results and discussions and section 7 finally concludes this paper.

2. Motivation

Several deep web clustering approaches have been proposed in the literature. In this section we discuss the limitations observed in the existing works that motivated us to do this research work. In the existing works limitations are observed from two perspectives. Both are related to the intrinsic property of clustering algorithms.

Existing clustering approaches proposed for deep web clustering are hard clustering that perform hard assignment. In hard clustering approach, a document can only belong to a single cluster. As documents comprise of multi-topics, therefore hard clustering is not a suitable option for identifying its topics. Therefore today soft clustering techniques are acquiring high

attention for document classification. Similarly due to sparse nature of deep web source's content, a deep web source may belong to multiple domains. So here we argue that hard clustering techniques are not suitable for them.

Existing clustering approaches do not encounter semantics found in a deep web source. In their paper, [6] proposes two approaches to cluster deep web sources i.e. CAFC-C and CAFC-CH. The former generates homogeneous clusters with low entropy and high F-measures. Thus it is considered to be very effective in discriminating online deep web databases. But the approach has some intrinsic limitations due to the formal k-means clustering approach. With respect to k-means clustering approach, quality of the resultant clusters is dependent on the selection of the initial seeds. Thus the CAFC-C approach doesn't work well when there is high heterogeneity in vocabulary and when explicit domains share a common vocabulary. So here it is needed to have some other discriminating features to have an effective clustering for all kinds of scenarios. The situation is tackled by introducing hub induced similarity based clustering CAFC-CH. In this approach page contents are also considered for cluster discrimination.

This method effectively works for both the aforementioned scenarios. Further for a good seed selection, hyperlinks towards and from are considered. This approach has still encounter shortcomings regarding two scenarios: back-links are not available for every source and vector space model performs keyword based matching. Thus no procedure for semantics based relevance among the vocabulary is adopted.

Existing clustering methods as discussed so far are useful because they save labeling cost and are not time consuming but they have till now ignored the semantics of web documents which is important for web based clustering approaches. A deep web classification approach that considered semantics [10], builds ontology for each category domain, which is time consuming and ontology is developed for limited web interface models.

3. Related Work

Several deep web classification and clustering techniques have been proposed before a detailed survey of which can be found in [11]. Here we briefly discuss existing clustering approaches in order to have a clear understanding regarding the motivation towards our approach.

In order to cluster deep web sources, [5] proposes model-differentiation as a new objective function. The approach allows principled statistical measure for determining cluster homogeneity by deriving a new similarity measure for the HAC algorithm. Pre-clustering and post-clustering techniques are designed. Then statistical hypothesis testing is performed. The major limitation of this approach is that it encounters only multi-attribute query interface forms. However our proposed method clusters both single and multi-attribute query interface forms.

A new similarity computing algorithm "*literal and semantic based similarity computing (LSSC)*" has been proposed in [7] to compute similarity among deep web query interfaces. Further after computing similarity, NQ clustering is performed to cluster deep web query interfaces based on similarity measure. A new representation to the query interface is assigned: form term and function term. The former describes the literal information in the form that is ultimately used to describe corresponding controls, and this information can be navigated by search engines; the later describes the control information that is out of reach of search engines but this information is used to cluster web forms. The common attributes of a domain are taken as features of that domain. After the integration step of query interfaces, every cluster is matched with these features to appoint clusters to their corresponding domains. Again this approach encounters only multi-attribute query domains with high precision and recall.

A graph based representation of deep web is performed in [8] having multiple heterogeneous relationships. The nodes denote query interface form; the edges are the relation between the relevant query interfaces, and the relative weights are the similarity between them. Thus the whole form-set is represented in the form of weighted undirected graph. The weight of the edge is measured by matching degree between schemas of two attribute sets. For schema matching instead of using binary value logic, fuzzy set theory is used. Finally the extracted feature set is clustered using fuzzy clustering method.

4. Proposed Clustering Approach

In view of above limitations, our research work addresses two major issues: 1) soft assignment of deep web sources into

clusters and 2) semantic clustering of deep web sources. For both the issues we propose a clustering approach based on directed probabilistic topic model i.e. latent dirichlet allocation (LDA). LDA is a graphical model that provides a way for automatically discovering topics from a text corpus. Graphical models have long been used effectively for topic based probabilistic modeling [12]. Among them, models based on latent variables have proved to be highly effective in capturing the hidden structures in the data. Similarly LDA is a popular generative probabilistic model for collections of text corpora and other discrete data. LDA represents documents as being comprised of a number of topics. Each topic further comprise of words having certain probabilities. Here in this section we provide a brief overview of working of LDA. A detailed explanation can be found in [9][12].We first explain LDA model and then we shed light on one of its inference methods.

4.1 LDA Model

In order to generate a document through LDA model, for each document d_i in a corpus C following steps are performed:

- The number of words in a document is estimated by Poisson distribution denoted as N, thus for each interface

$$N \sim Poisson\,(\xi) \qquad (1)$$

- A topic mixture for the document is selected. This is done through a dirichlet distribution over a fixed set of Z topics i.e.

$$\ominus_{interface} \sim Dir\,(\alpha) \qquad (2)$$

- Each word W_i is generated in the document by following two steps:
  - Pick a topic using multinomial distribution

$$Z_i \sim Multinomial\,(\ominus_{interface}) \qquad (3.1)$$

  - Using topic's multinomial distribution, generate the word from a multinomial probability conditioned on topic Z_i

$$Probability\,(W_i / Z_i, \beta) \qquad (3.2)$$

A collection of documents is generated through this model. A mathematical definition for the above distributions is provided in the problem formulation section

4.2 Inference

Next to the document-topic modeling is the inference phase. In this phase various probability distributions for documents, topics and words are learnt. These distributions include word probabilities for the set of topics and topic mixture for each document. The inference techniques used are variational Bayes approximation of the posterior distribution, Gibbs sampling and expectation propagation. In this paper we use Gibbs sampling. The process starts by randomly assigning each word w_i in the document d_i to one of the z_i topics. To improve on this random assignment the following step is iteratively performed a large number of times by assigning new topic to the word each time.

$$Probability\,(z_i \mid d_i) * Probability\,(w_i \mid z_i) \qquad (4)$$

Where Probability ($z_i \mid d_i$) determines the proportion of those words in document d_i which are assigned to topic z_i and Probability ($w_i \mid z_i$) determines the proportion of the assignments to topic z_i from all over documents that come from this word w_i.

4.3 Content Representative Extraction and Pruning

In order to cluster a deep web source we extract a sample from the whole content which must be an excellent discriminator of the source. Such a sample that adequately represents all the key characteristics of a data source leads to effective clustering. Therefore we extract a broader set of metadata associated with the deep web source including:

- Textual content on the form page
- The attribute labels of the form i.e. query schema.

Such extracted content has the ability to cluster both simple/advance search query interfaces. Thus our approach provides highest deep web coverage as compared to the rest of the approaches.

In view of above, the sample which we select in our approach is the Form-Page (FP) contents. A FP comprises of two individual feature spaces i.e. Page Contents (PC) and Form Contents (FC).

In order to extract FP contents, HTML page of the deep web source is parsed, and two feature spaces FC and PC are computed. The standard stop word removal is used for both form and page feature space. Then less frequent terms those occur three times or less in whole collection are removed. TF-IDF (term frequency/inverse document frequency is used to calculate the weights of terms of forms and pages.

5. Problem Formulation

In this section we formally formulate our research problem. First of all we describe the related conception of our proposed technique. Then we present the pseudo-code for our technique

5.1 Related Conception

5.1.1 Definition 1 (Document, Words, Text Corpora)

A single document comprises of words or terms represented as $di : wi = \{w1, w2, w3, \ldots\ldots, wN\}$. A document collection also called as text corpora is represented as $C = \{d1, d2, d3, \ldots\ldots, dM\}$.

5.1.2 Definition 2 (Topic)

A topic Z is defined as a probability distribution over words in a vocabulary corpus. Formally $Z = \{Prob(w1), Prob(w2), \ldots., Prob(wn)\}$.

5.1.3 Definition 3 (Dirichlet Distribution)

The Dirichlet distribution denoted as $Dir(\alpha)$, is a family of continuous multivariate probability distributions that are parameterized by the vector α of positive real's. In Bayesian statistics dirichlet distribution is the multivariate generalization of the beta distribution and the conjugate prior of the categorical distribution and multinomial distribution. That is, its probability density function returns the belief that the probabilities of E rival events are xi given that each event has been observed $\alpha i - 1$ times.[1]

5.1.4 Definition 4 (Multinomial Distribution)

The multinomial distribution is the generalization of binomial distribution, in which each independent trial has k possible discrete outcomes instead of two outcomes. The process consists of 'n' repeated trials where for each trial the probability that a particular outcome will occur is constant. The outcome of one trial doesn't affect the outcomes of the other trials. Multinomial distribution is used to model random categorical data.

5.2 DWSemClust

We propose DWSemClust which takes the parsed Form-Page as input for LDA. LDA can capture the semantics of forms and pages in an unstructured way by making soft clusters of data through latent topic layer. Latent topic layer allow documents that are composed of different topics to belong to more than one cluster. Basic unit of discrete data is a word. Words are unique items in a vocabulary denoted as w and combination of words is called document d. A document contains N words denoted as $d = \{w1, w2, \ldots, wN\}$ and corpus is the collection of all documents denoted as $C = \{d1, d2, ..., dM\}$ which shows that corpus contains M documents. Topic layer is denoted by $Z = \{z1, z2, \ldots, zi\}$ between the documents and words in the documents, where zi represent latent topic a document vector d words w.

This layer is used to capture the semantic relationship that considers the synonymy and polysemy of words. For each FormCon, PageCont iterated 'M' times. Select èFormCon_PageCon from hidden parameters α. For each WFormCon_PageCon iterate 'N' times for each document. ZFormCon_PageCon is selected from $\ominus$ FormCon_PageCon. WFormCon_PageCon is the observable variable in probability of (W FormCon_PageCon | Z FormCon_PageCon , β). Finally we obtain semantic clusters from the whole process

[1] Wikipedia: http://en.wikipedia.org/wiki/Generative_model

	Algorithm 1: DWSemClust
Input:	Set of searchable deep web interfaces
Output:	Clusters
1.	*FormCon_PageCon = Parser(Input)*
2.	*D1 = LDA(FormCon_PageCon)*
3.	*For each (FormCon_PageCon* [*1…M*]) *do*
4.	Select ($\ominus_{FormCon\_PageCon}$) ~ *Dir* (α)
5.	For each term *W FormCon_PageCon* [1…*N*] do
6.	Select a topic $Z_{FormCon\_PageCon}$ ~ *Multinomial* ($\ominus_{FormCon\_PageCon}$)
7.	Select a word $W_{FormCon\_PageCon}$ from probability ($W_{FormCon\_PageCon}$|$Z_{FormCon\_PageCon}$, β) // a multinomial probability conditioned on the topic $Z_{FormCon\_PageCon}$
	Algorithm 2: Parser
Input:	Deep web searchable interface
Output:	Form contents, Page contents
1.	*FP = StopwordRemove* (*Input*)
2.	*FormCon_PageCon = NoiseRemove* (*FP*)

Table 1. Proposed Algorithm DWSEMCLUST

6. Experimental Evaluation

In this section, we present the results from the experiments using our DWSemClust clustering approach. The presented experiments were designed as comparisons of the proposed method with selected traditional deep web clustering approaches i.e. CAFC_C and CAFC_CH [5]. The observations were based on the performance measures of F-measure and Entropy. Through experiments we proved that our technique works notably well for both measures.

6.1 Dataset

In order to evaluate the performance of our approach we tested it over dataset of TEL-8, UIUC Web integration repository [11]. The repository contains web search forms of 447 sources originally classified into 8 domains. The term TEL-8 refers to eight different web source domains belonging to three major categories (Travel, Entertainment and Living). Travel group is related to car rentals, hotels and airfares. Entertainment group has books, movies and music records interfaces. Finally the living group contains jobs and automobiles related query interfaces. Currently some of the form pages are not updated, hence not available for the experiment. Finally we collected 259 search interfaces for our experiment. The description of the dataset is given below in table 2.

6.2 Performance Measure

In order to evaluate the performance of our approach we used two kinds of measures i.e. F-measure and entropy. The F-measure is a combined measure of precision and recall represented as weighted harmonic mean. Precision and recall can be computed as:

$$Precision = TP/(TP + FP) \quad (5)$$

$$Recall = TP/(TP + FN) \quad (6)$$

Group	Domain	Query-able Forms
Travel	Airfare	34
	Hotel	26
	Car rental	17
Entertainment	Books	42
	Movies	41
	Music	35
Living	Jobs	25
	Automobiles	39

Table 2. Dataset Description

In above, TP (true positive) denotes those members of the set that has been clustered correctly. FN (false negative) denotes those members of the set that belongs to a particular domain but is falsely clustered in another domain. FP (false positive) denotes those members of the set that are falsely clustered into a domain. There is tradeoff among precision and recall, improving one leads to negative effect on the other. To overcome this tradeoff, F-measure is defined as:

$$F\text{-}measure = (2*Precision*Recall) / (Precision + Recall) \quad (7)$$

A high F-measure means that both recall and precision are high.

Entropy is a measure of disorder in clusters. Cluster performance increases as the entropy decreases. For every cluster the possibility of occurrence that a member of cluster '*j*' belongs to a class '*k*' is computed. Entropy is found as:

$$Entropy_j = -\Sigma\, pjk\ log\ (pjk) \quad (8)$$

6.3 Parameter Estimation

Hyper parameters α and β can be optimized through Gibbs sampling algorithm [12] or Expectation Maximization (EM) method [13]. Gibbs sampling algorithm is used rather than EM algorithm as EM is computationally inefficient and vulnerable to local maxima [8]. Hyper parameters need optimization as some topic models related to different applications are sensitive to these parameters. But in our case, topic is not sensitive to hyper parameters. In our experiments for 8 topics *z* the hyper parameters values for α and β are respectively $50/z$ (α is commonly set as 50) and .01 (an increase in the value of β will result in sparse topics while decrease will result in dense topics). The value of topics *z* is set as respect to our dataset used i.e. 8. Topic optimization is usually dependent on size of the dataset, as a small dataset will usually be optimized at a small number of topics, as compared to a large dataset.

6.4 Results and Discussion

In order to obtain accurate results, we computed entropy for 25 random generations of LDA clusters. F-measure was computed for 10 random generations of LDA clusters. The values were computed for both content representatives i.e. FC (form contents) and FP (form-page contents). Figure 1 graphically represents our observations. Average entropy of DWSemClust with form and pages in both scenarios is lower as compared to the CAFC. The lower the entropy the less the disorder shows that semantically clustered deep web sources are more precise. Entropy curve for DWSemClust show that our proposed approach is both efficient and more stable as compared to CAFC for multiple generations. This is an important aspect in our work.

Similarly in Figure 2 a high value for F-measure results in better precision and recall. We see here that average F-measure of DWSemClust with forms and pages in both scenarios is higher than the CAFC, which shows the accurate recognition of deep web sources. Also we see that FP as content representative also helps improve the clustering results.

In Figure 3 and 4, a comparison of CAFC_C, CAFC_CH and DWSemClust are shown in terms of entropy and F-measure. CAFC_C use random selection of documents and CAFC_CH use the hub induced similarity as a preprocessing step. First of

all hubs are generated and the number of clusters are selected and then the algorithm is run for of CAFC which clusters the sources among all methods our proposed DWSemClust perform well in both form and in form-page scenarios.

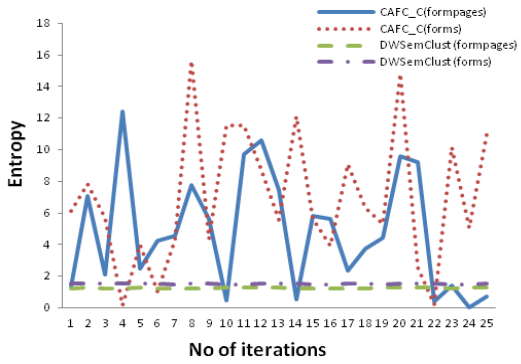

Figure 1. Effect of different number of iterations on CAFC_C and DWSemClust with forms and formpages contents in terms of entropy

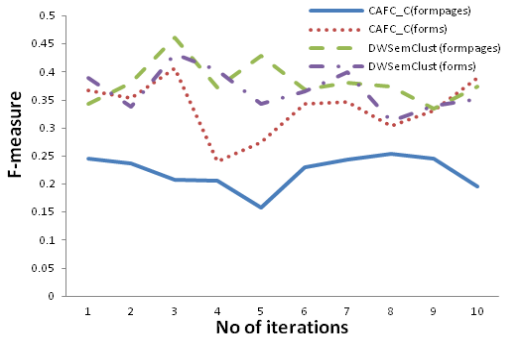

Figure 2. Effect of different number of iterations on CAFC_C and DWSemClust with forms and formpages contents in terms of F-measure

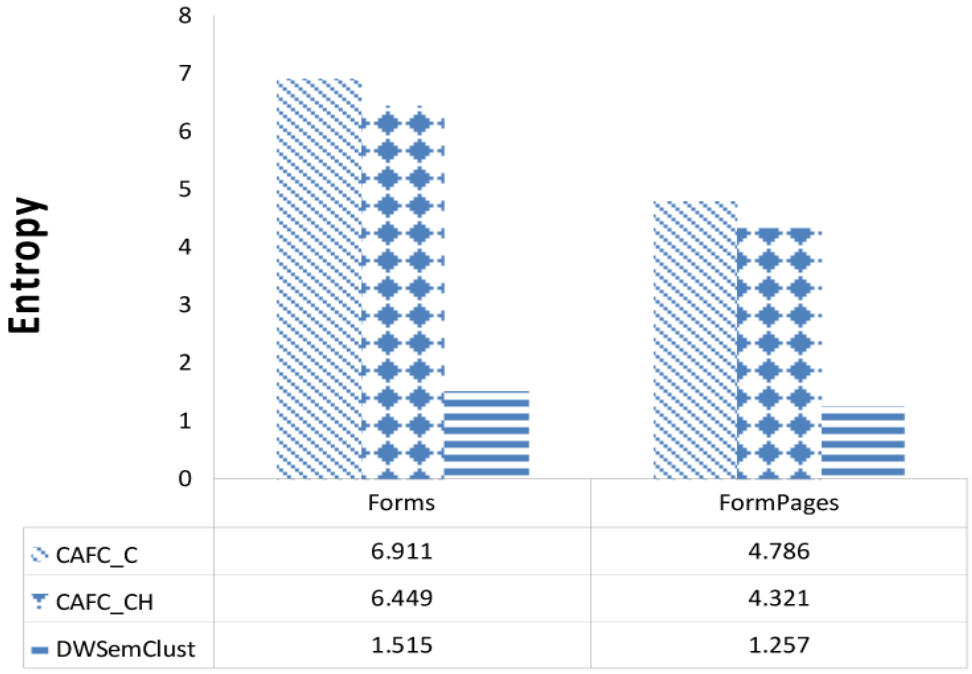

Figure 3. Entropy comparison of CAF_C, CAFC_CH and DWSemClust

all hubs are generated and the number of clusters are selected and then the algorithm is run for of CAFC which clusters the sources among all methods our proposed DWSemClust perform well in both form and in form-page scenarios.

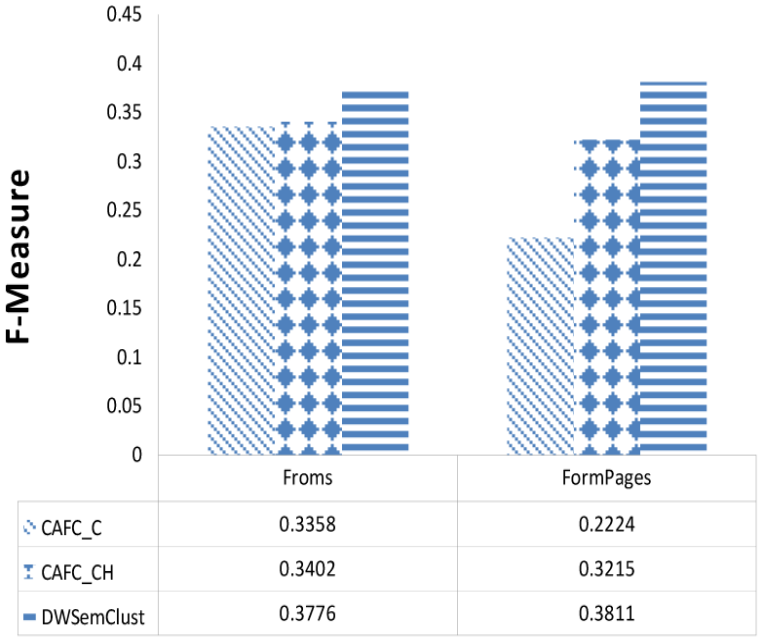

Figure 4. F-measure comparison of CAF_C, CAFC_CH and DWSemClust

Entropy value of CAFC_C and CAFC_CH is higher than the DWSemClust as entropy value increases the performance of clusters decreases. As F-measure value increases the performance of cluster also increases. DWSemClust have more F-measure value.

Finally we also observe the topic distribution actually performed by above experiments. Table 3 shows the word clusters for the defined topics derived through DWSemClust by considering the semantic similarities. The words associated with each topic for DWSemClust are strongly semantically related and they make compact topics in the sense of conveying a semantic summary of a specific domain. We see that our approach effectively handles sparse topic distribution of deep web sources.

7. Conclusion

The result shows that semantics are useful for clustering heterogeneous deep Web sources. Our proposed semantics-based technique DWSemClust is more efficient than existing non-semantics based techniques. It takes less time to give the clustering results. As LDA produce soft clusters it assigns probability to each document for each cluster. Hence, DWSemClust is suitable for the scenario where the sources are sparsely distributed over the web.

References

[1] The Deep Web. (2000). Surfacing hidden value. Accessible at http://brightplanet.com.

[2] Madhavan, J., Cohen, S., Dong, X. L., Halevy, A. Y., Jeffery S. R., Ko, D., Yu, C.(2007). Web scale data integration: You can afford to pay as you go. *In*: Proceedings of Conference on *Innovative Data Systems Research* (CIDR), p. 342–350.

[3] Chang, K. C. C., He, B., Li, C., Patel, M., Zhang, Z. (2004). Structured databases on the web: Observations and Implications. *In:* Proceedings of *International Conference on Management of Data* (ACM SIGMOD), p. 61–70.

[4] He, B. Tao, T. K., Chang, K. C. C. (2004). Organizing structured web sources by query schemas: a clustering approach. *In*: Proceedings of Conference on *Information and Knowledge Management* (CIKM), p. 22–31.

[5] Barbosa, L., Freire, J. and Silva, A. (2007). Organizing hidden-web databases by clustering visible web documents. *In:* Proceedings of International Conference on Data Engineering (ICDE), p. 326–335.

[6] Lin, P., Du, Y., Tan, X., Lv, C. (2008). Research on Automatic Classification for Deep Web Query Interfaces. *In: International Symposium on Information Processing* (ISIP), p. 313–317 .

[7] Zhao, P., Huang, L., Fang, W., Cui, Z.(2008). Organizing Structured Deep Web by Clustering Query Interfaces Link Graph. *In*: *Lecture Notes in Computer Science*, 5139, p. 683–690.

[8] Blei, D., M., Ng, Y., A., Jordan, M., I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3, p. 993-1022.

[9] Xu, H., Hau, X., Wang, S., Hu, Y. (2007). A Method of Deep Web Classification. *In*: Proceedings of International Conference on *Machine Learning and Cybernetics* (ICMLC), p. 4009–4014.

[10] Daud, A., Li, J., Zhou L. and Muhammad, F. (2010). Knowledge Discovery through Directed Probabilistic Topic Model: a Survey. *Journal of Frontiers of Computer Science in China* (FCS), Vol. 4 (2) 280-301.

[11] The UIUC Web integration repository http://metaquerier.cs.uiuc.edu/repository.

[12] Griffiths, T., L. and Steyvers, M. (2004). Finding scientific topics. *In*: Proceedings of National Academy of Sciences (NAS), p. 5228–5235 .

[13] Hofmann, T.(1999). Probabilistic latent semantic analysis. *In:* Proceedings of 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI), Stockholm, Sweden, p. 289-296 .

[14] Raghavan, S., Garcia-Molina, H.(2001): Crawling the Hidden Web. *In*: VLDB, p. 129–138 .

[15] Hess, A., Kushmerick, N.(2003): Automatically attaching semantic metadata to web services. *In:* II Web, p. 111–116

[16] Dong, Y., Q., Li, Q., Z., Ding Y., H.(2010): A query interface matching approach based on extended evidence theory for Deep Web. *Journal of Computer Science and Technology*, 25 (3) .

[17] Salton, G., Wong, A., Yang, C., S. (1975). A vector space model for automatic indexing, *Journal of CACM*, 18 (11) 613–620

[18] Steinbach, M., Karypis, G., Kumar, V. (2000). A comparison of document clustering techniques. *In:* KDD Workshop on Text Mining

[19] Larsen, B., Aone, C. (1999). Fast and effective text mining using linear-time document clustering. *In*: proceedings of KDD, p. 16–22 .

[20] Le, H., Conrad, S. (2010). Classifying Structured Web Sources Using Support Vector Machine and Aggressive Feature selection. *In:* Lecture Notes in Business Information Processing, 45, p. 270–282

[21] Xian, X., Zhao, P., Fang, W., Xin, J., Cui, Z. Automatic Classification of Deep Web Databases with Simple Query Interfaces. In: proceedings of International Conference on Industrial Machatronics and Automation (ICIMA), p. 85–88

1st Topic Music Record		2nd Topic Airfares		3rd Topic Movies		4th Topic Hotels	
Words	**Probabilities**	**Words**	**Probabilities**	**Words**	**Probabilities**	**Words**	**Probabilities**
Books	0.0360	City	0.0664	Prices	0.0378	Deals	0.0377
Music	0.0287	Airport	0.0658	Movies	0.0341	Help	0.0322
Books	0.0253	Flight	0.0553	Rewards	0.0317	Travel	0.0299
News	0.0200	Child	0.0442	Sort	0.0249	Document	0.0283
Online	0.0193	Name	0.0378	Search	0.0244	Information	0.0259
Contact	0.0173	Infant	0.0361	Artist	0.0207	Vacation	0.0252
Shop	0.0167	Flights	0.0332	Miles	0.0195	Hotel	0.0228
Store	0.0167	Hotel	0.0279	Reservation	0.0183	Save	0.0228
DVD	0.0160	Include	0.0198	Options	0.0177	Cheap	0.0220
Gift	0.0153	Options	0.0180	View	0.0171	Map	0.0212
Toys	0.0153	Lap	0.0169	Confirmation	0.0171	Sign	0.0204
USD	0.0133	Children	0.0163	Ticket	0.0164	Privacy	0.0181
Games	0.0133	Seat	0.0163	Close	0.0152	Destination	0.0181
Browse	0.0120	Class	0.0157	wyndham	0.0152	Cheapfares	0.0173
Prices	0.0113	Miles	0.0146	Special	0.0140	hotels	0.0165

5th Topic Jobs		6th Topic Automobiles		7th Topic Car Rental		8th Topic Books	
Words	**Probabilities**	**Words**	**Probabilities**	**Words**	**Probabilities**	**Words**	**Probabilities**
Jobs	0.1201	Cars	0.0553	Car	0.0778	Title	0.0619
Pound	0.0497	Chevrolet	0.0281	Time	0.0589	Advanced	0.0503
Job	0.0274	Ford	0.0281	Code	0.0531	Price	0.0319
Saving	0.0249	Toyota	0.0281	Select	0.0479	Keyword	0.0300
Details	0.0207	Car	0.0281	Travel	0.0392	Author	0.0290
Price	0.0197	Quantity	0.0256	Pick	0.0371	Keywords	0.0281
summary	0.0171	Vehicle	0.0239	Drop	0.0349	Model	0.0242
Sales	0.0150	Nissan	0.0215	Ages	0.0327	Isbn	0.0242
Health	0.0135	Honda	0.0198	Date	0.0327	Below	0.0223
Business	0.0129	Cyl	0.0189	Address	0.0283	Results	0.0213
Manager	0.0124	Bmw	0.0157	Rental	0.0247	Abc	0.0184
London	0.0119	Dodge	0.0157	Location	0.0233	Category	0.0145
Care	0.0114	Hyundai	0.0157	Zip	0.0218	Format	0.0136
Application	0.0109	Mercedes	0.0140	City	0.0211	Fields	0.0136
South	0.0098	Benz	0.0140	Airport	0.0189	Exact	0.0126

Table 3. An Illustration Of Eight Discovered Topics With Each Topic Shown With The Top 15 Words And Their Probabilities