CSIS: Cloud Service Identification System

Siddharth Arun, Aakash Chandrasekaran, Prakash P
Department of Computer Science and Engineering, Amrita School of Engineering, India

Article Info	ABSTRACT
	To meet the need of the computational power, most of the users may go for a cloud based services for its scalability, flexibility and reliability. Cloud services have become an integral part of IT and analytical enterprises. Owing to increase in necessity of commercial cloud products being readily available, it has become extremely difficult for users to identify suitable cloud services. This paper proposes the recommender system precisely designed for the discovery of cloud services. Though there is an exponential increase in demand for cloud services, the amount of research done in this particular field is abysmal. Cloud Service Identification System (CSIS) crawls through Internet, identifies cloud services and stores them in a database. The user's search query is processed and recommends cloud services accurately.

Corresponding Author:

Siddharth Arun,
Department of Computer Science and Engineering,
Amrita School of Engineering,
Amrita Vishwa Vidyapeetham, India.
Email: cb.en.u4cse13052@cb.student.amrita.edu

1. INTRODUCTION

With the onset of innovation of the cloud services computations in local computers could be a thing of the past. It would be brought under one roof operated by third-party computers and storage utilities. Cloud Computing is a model used to implement flexible, beneficial, high on-demand network access to a shared pool of computing resources, that can be quickly provided with minimal effort, for e.g., applications, servers, storages, networks and other services [1], [2].

Since there is a multitude of options in the cloud market, customers are often confused in choosing an appropriate product due to the lack of knowledge in this domain. [3] Salesforce, an American-based SaaS company famous for Customer Relationship Management (CRM) applications, offers a Simple Object Access Protocol (SOAP) / Representational State Transfer (REST) Web service API that enables integration with other systems. Global SaaS software revenues are forecasted to reach $106B in 2016, increasing 21% over projected 2015 spending levels [4].In spite of this growth, there is no actual catalog of cloud products. This has made the users search for potential cloud products that meet their requirements.

The motive of this work is to help customers to find an ideal cloud service and other related services based on their requirements. Cloud Service Identification System (CSIS) identifies cloud service and recommends them to customers.

This paper is categorized in the following way. Section 2 discusses the problem statement and provides a solution for it. Section 3 provides an outlook of the system. Section 4 portrays the architecture of the system and its important components. Section 5 outlines the implementation and results of the CSIS system. Section 6 concludes the problem.

Generally, Customers resort to search engines initially to find services. Normal search engines are not built to find services suitable to customer's needs and prioritize them. Other than search

engines, Traditional marketing; has a proven success rate which refers to any type of promotion, advertising or campaign that has been in use by companies for years.

The standard search engines present the results in a layout based on text which is challenging for users to judge their service without being thoroughly examined. This proposed system is trying to find a solution for the above mentioned problem and help the customers to find suitable services based on their requirements.

In 2012, a peer-peer, business software review platform was founded called "G2 crowd". This vendor although focuses on aggregating user reviews for business software, but the shortcoming of this software is manual addition of new cloud services at every stage.

In 2005, the Salesforce AppExchange was launched, it is an online application marketplace of various business applications and consulting partners that provide solutions for companies operating with various needs and various industries. Even though they provide technically sound search engine AppExchange requires the customers to have a common knowledge about the software application, hence it is not suitable for new users.

The CSIS system addresses the issue by using hybrid recommendation system which is a combination of both content-based and collaborative recommendation system, in recommending suitable cloud services based on user's requirements [5]. This system overcomes the aforementioned problems of the system.

2.   RESEARCH METHOD

The main motivation of the CSIS system is to help customers in identification of cloud services and suggest other relevant services which suits their needs. The system is categorized as five different components:
a. Crawler is a bot which automatically browses the Internet, generally for the purpose of Web indexing.
b. Identifier is used to identify if a web page is genuine cloud service based on its service score.
c. Indexer is used to register a cloud service which is identified.
d. Search Engine is used to browse for cloud computing products.
e. Recommender is used to recommend both content-based and collaborative recommendations.

From the user's outlook, they will visit the CSIS webpage and register for an account. Once the account is registered, they are provided with search utilities in which they search for cloud services and rate them on a scale of one to ten. Then the user will be recommended a list of cloud services based on their requirements. Some of the real world scenarios, of how the system can be used:
a. Customers who are using SaaS products and not satisfied, due to poor quality visit the CSIS webpage and rate the cloud product which they have been using lately, then they are recommended other similar cloud services based on their ratings and other user's ratings.
b. An individual who is new to cloud computing and wishes to use a cloud product. They visit the webpage of CSIS system and enter a search query in search of cloud products. A list of cloud products is displayed according to their ranks, to the user. The user would choose a product based on their requirements, once the user chooses a product, the CSIS system would suggest other products that integrate with the current chosen service and the one with the highest rating is displayed.

2.1. Architecture

CSIS system architecture is the conceptual model that defines the structure, behavior, and more views of a system. A basic overview of the CSIS system architecture is displayed in Figure 1.
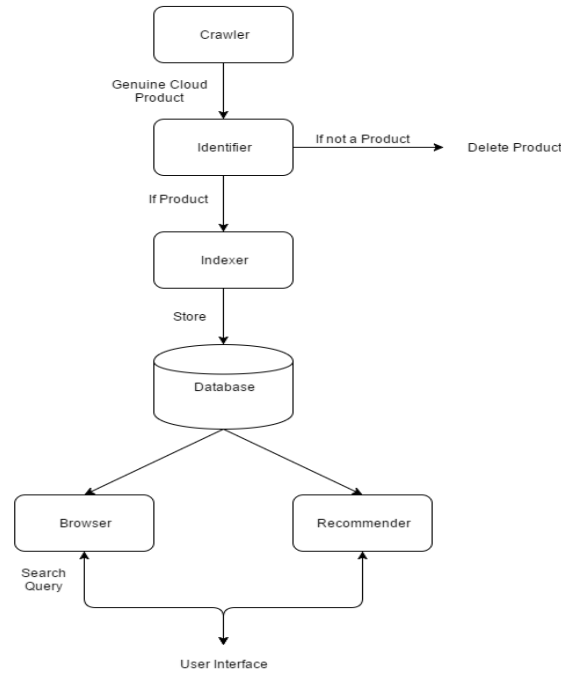
Figure 1. CSIS Architecture

2.1.1. Crawler

The purpose of crawler is to find the content of unknown websites that can be fed as an input to the Identifier. It is expected to crawl more number of webpages as majority of the websites visited won't be a genuine cloud service website, hence it requires more time before an actual service is identified. Therefore, to improve the efficiency of crawling, distributed crawlers are used. Initially a list of blogs and directories are compiled with web links of cloud services, before crawling to the Internet, as it would be unwise to wildly follow URL in search of cloud services. There is also a possibility of repetitive downloading of same websites while crawling the Internet. To prevent this from happening, the crawler caches a copy of the websites visited. This may result in an overload of cache data; hence the cache is cleared when the cloud product is either identified or registered.

2.1.2. Identifier

The goal of Identifier is to identify a given webpage as a genuine cloud product and provide a score based on their level of service. This helps the systems to crawl over Internet in search of cloud services which can be searched and recommended. Individually it would be difficult to check a cloud service as majority of cloud services requires certification to access. Rather, universally accessible homepage is examined if it a potential cloud service. The basic fundamental is to compare the homepage crawled with a potential website and find the similarity between.

To accomplish the similarity, the CSIS system is initially fed with various URL of known genuine cloud service. When the system is activated, it visits the homepage of every URL separately and stores the key words which appear, along with the frequency of the words across all cloud services. Any word with count of one is subtracted as it holds no importance.

High frequency words are most interested but that doesn't mean low frequency words can be ignored as some may hold importance in niche market. Score of a word is calculated by using Equation 1.

$$\text{Score} = \log (\text{Frequency of Word}) \tag{1}$$

The list of words and respective scores will be examined with the profile of a potential cloud product. The random webpage is compared to this profile to resolve the possibility that given webpage is a cloud product. This comparison is done by computing the similarity between random webpage (w) and genuine cloud service (g). Both webpage (a) and genuine cloud service (b) can be represented as term frequency–inverse document frequency (TF-IDF) vectors $\vec{W}_a$ and $\vec{W}_b$ of weights (scores). These two

vectorsare considered as two vectors in k-*dimensional* space where k =|P$_{ab}$|, where P$_{ab}$ is the set of all items co-rated by both the webpages [6].

The similarity measure sim (a, b) is given by Equation 2.

$$sim(a,b) = \frac{\sum_{p \in P_{a,p}}(W_{a,p}W_{b,p})}{\sqrt{\sum_{p \in P_{ab}}(W_{a,p}^2)}\sqrt{\sum_{p \in P_{ab}}(W_{b,p}^2)}} \tag{2}$$

To calculate the similarity measure, the list with the seeded cloud services is compared with the ideal profile. When Equation 3 is satisfied by score w, a cloud product is detected where q is set of all cloud product scores.

$$w = \acute{q} \pm 4 \tag{3}$$

2.1.3. Indexer

An Indexer indexes a website if it is genuine cloud product and stores it in the database of the CSIS System. Figure 2 depicts the Architecture of Indexer.
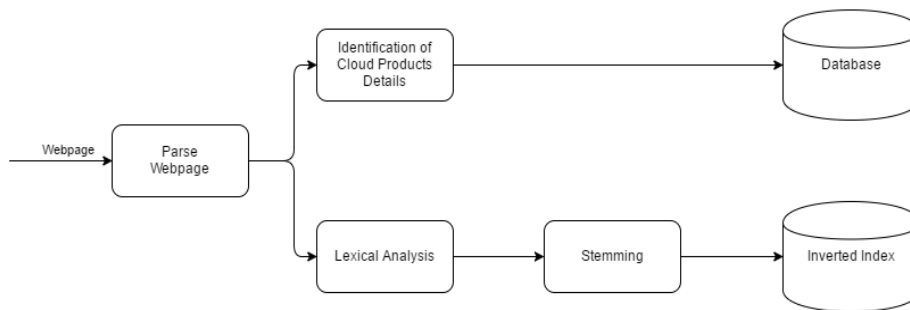

Figure 2. Architecture of Indexer

Initially the Indexer parses the webpage and the fundamental details such as product name, image and description of the website is extracted and stored in database with unique index number (ID) which is used to identify the cloud service throughout CSIS.

On the parsed text, Lexical Analysis is performed and each individual word is identified and Porter Stemming Algorithm is used to stem each word [7] and the Document frequency (*df*) is determined for the page.

Using Equation (4), the inverted document frequency (*idf*) is determined on iterating basis, where T is the total number of cloud products and *idf* is determined using document frequency(*df*) for a particular term k [8].

$$idf_k = log\frac{T}{df_k} \tag{4}$$

The term index ID and corresponding frequency are stored as an item, along with the idf, in an inverted index, as in Figure 3.
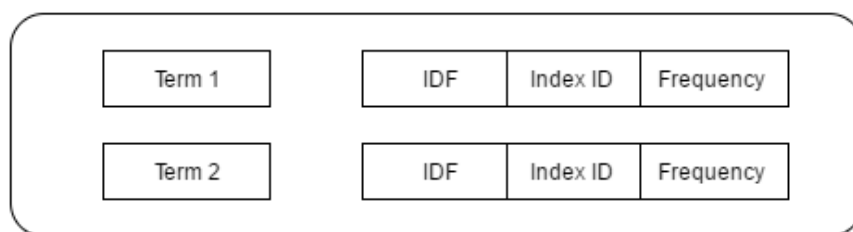

Figure 3. Indexer Posting

2.1.4. Search Engine

It helps the user to easily find cloud products stored in the database based on specific query or term which includes either the name of the product or specific feature they require.

When the search engine receives a query, Lexical Analysis is done and the resulting terms are stemmed. Using Equation 5, the *idf* and the frequency f, of the term is calculated for every term in the document.

$$f - idf = tf_{t,d} \cdot idf_t \tag{5}$$

To rank the results, Vector Space Model is used where the search query, r is compared with the calculated *tf-idf* weight w of the document d. For instance, $tw_{x,y}$ is the weight of the xth word in document *y* [8].

$$(d_y, r) = \frac{d_y \cdot r}{\|d_y\| \|r\|} = \frac{\sum_{x=1}^{N} w_{x,y} w_{x,r}}{\sqrt{\sum_{x=1}^{N} w^2_{x,y}} \sqrt{\sum_{x=1}^{N} w^2_{x,r}}} \tag{6}$$

Provided a large number of cloud services, if the search query includes any of the cloud service's name it is shown first of the results.

2.1.5. Recommender

It is the primary part of the CSIS System which uses a hybrid recommendation approach which is a combination of both content-based as well as collaborative recommendations. The hybrid approach is used so as to remove the drawbacks of using other approaches individually [6].

In content-based recommendation approach, user uses the search query to find the cloud product, it also recommends other products that are related to the query. This may help the user to find cloud service based on their requirement.

As stated before, it is common that user may be highly interested with the product based on their integrity with another product, hence it should also be recommended. Without human interaction it is impossible to find whether a product is integrated with another product. Rather, it has been found that cloud service which has integration with another cloud service usually mentions it in the homepage. This cloud service is searched using search engine by using its name and corresponding results are recommended, ignoring itself. This approach may have limited perspective as some services may have complex descriptive names. Even though it is a simple method, it shows relevant results.

Unlike content-based approach, collaborative recommendation approach predicts the relevant items for the user based on other user's ratings [6]. To facilitate collaborative recommendation approach, the CSIS System enables the user to give ratings for the cloud services.

The first method of collaborative recommendation approach is that it recommends other cloud services that are highly rated by other users who has given high ratings for the same cloud service which is been viewed by the user.

The second method of collaborative recommendation approach is that it finds other users similar to the current user by using cosine similarity measure [9], mentioned in Equation 2.

When the recommendation is required, this similarity between the current user and other user is calculated, to increase the performance these similarities is calculated before and only recalculated periodically.

By combining the collaborative and content-based recommendation approach, the weighted average is determined for each recommendation. Initially, due to less number of users and ratings, content-based approach seems superior than collaborative approach, but as the user's population increases, the collaborative recommendation approach results will increase and will be most accurate [10].

3.   RESULTS AND ANALYSIS

For the backend implementation of CSIS system, Java with Node.js server is used. Nearly, 1000 cloud products have been sought out for and indexed and few alpha testers have used the system to create an initial profile of users for initial recommendations. Refer Figure 4 and Figure 5 for a screenshot of web based user interface for the CSIS system. During the testing of CSIS Identifier, twenty cloud services were randomly choosen with a further thirteen webpages covering different genres. For each particular webpage, corresponding score was calculated and verified to see if it satisfies Equation 3. The results of the experiment are shown in Table 1.

Table 1. Cloud Product Identifier Test Results

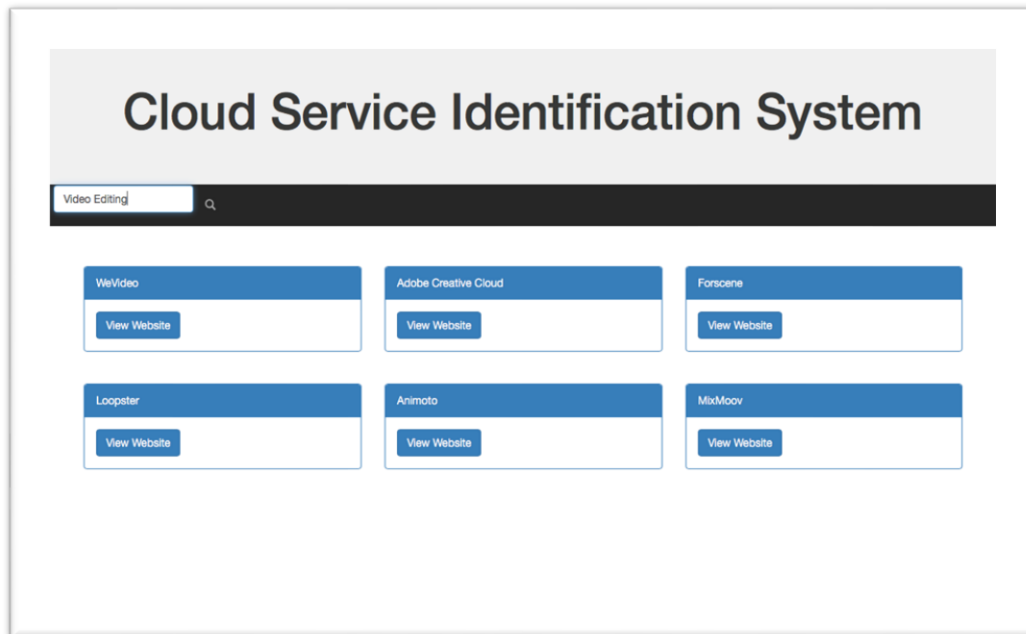| Type of the Website	Total number of Websites	Number identified as Cloud Product	Accuracy (%)
Cloud Product	20	16	80
Other	13	8	61.5
Total	33	24	72.73

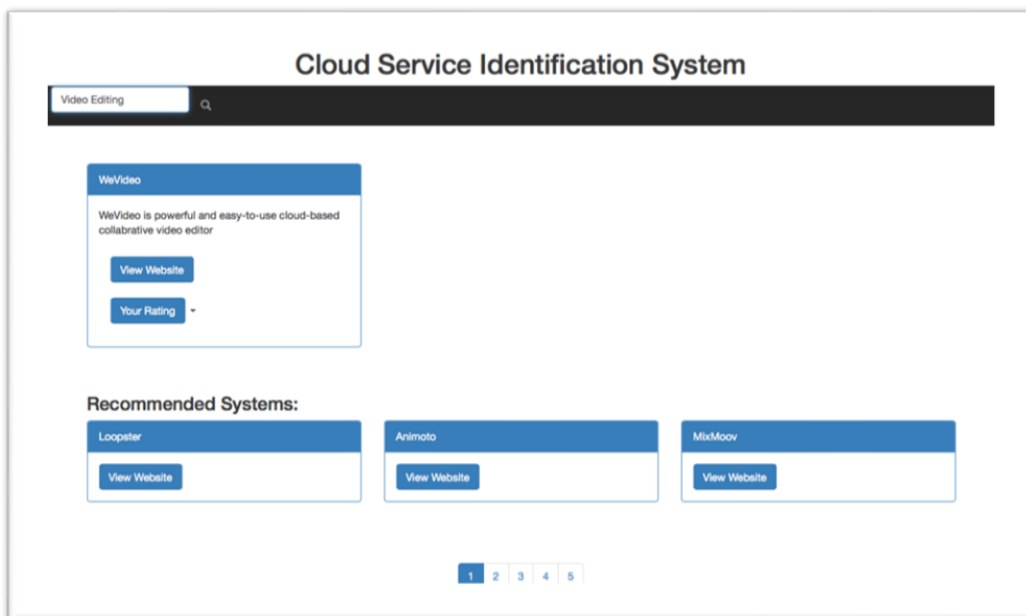Figure 4. CSIS User Interface

Figure 5. CSIS User Interface

The result of the experiment done, shows that the overall accuracy of the CSIS Identifier is 72.73%. Further work has to be done to increase the accuracy of the system.

The search engine has been tested using a wide variety of search terms. Its precision has been calculated using the Equation 7.

$$Precision = \frac{Number\ of\ items\ retrived}{Total\ number\ of\ items\ retrived} \quad\quad (7)$$

The results have shown that the precision of the search engine varies depending on the nature of the search query. Queries that contain terms that are commonly associated with all cloud products is a particular area in which the search engine is weak, e.g. storage, security, backup.

The system was tested by eight users with real world scenarios, where both experienced as well as non-experienced users were asked to use the system. Five of the users had no experience using cloud services while the remaining three had extensive experience. The results of the experiment are shown in Table 2.

Table 2. Recommender Test Results

User Type	Number of Recommendation results	Number of Useful Results	Precision	Average Precision
Inexperienced User	6	5	0.83	
	6	4	0.67	
	3	0	0.00	0.52
	5	3	0.60	
	4	2	0.50	
Experienced User	7	6	0.85	
	5	3	0.60	0.77
	8	7	0.87	

The result of the experiment show that, on average, the precision of the Recommender system is higher for experienced users than inexperienced users. This result is expected as experienced users would have been able to rate existing services that they use and therefore it would have recommended results derived from collaborative methods. For further improvement of the Recommender System, Cliques-based Data Smoothing Approach [11] or Clustering based approach [12] can be implemented, so that the precision will be improved and the system will be prevented from data scarcity.

4. CONCLUSION

This project helps to identify better cloud services for customers based on their needs in this market. The system also helps the users who are not familiar with cloud services by making it less intricate and complex. It has a high scope in the market and has a potential solution in discovery of new cloud products to the customers. To further improve the accuracy of Cloud Service Identification System (CSIS), clustering based approach can be implemented.

As of now, CSIS used to identify URL of a potential cloud service and ratings given by other users. To further improve the efficiency of the system, an extension can be added where in a cloud service can be identified according to their type of cloud services such as SaaS, PaaS, IaaS.

REFERENCES

[1]   Sangeetha K.S, Prakash P, "Big Data and Cloud: A survey", *Advantages in Intelligent Systems and Computing (AISC)*, 325, pp. 773-778, Springer India.
[2]   Shiva Jegan R.D., Vasudevan S.K., Abarna K., Prakash P., Srivathsan S., Gangothri V. "Cloud computing: A technical Gawk", (2014) *International Journal of Applied Engineering Research*, 9 (14), pp. 2539-2554.
[3]   Wadhe MV, Bharadi VA. Review: Services and Applications of Cloud Computing. *International Journal of Scientific & Engineering Research*. 2013 Dec 12:979-84.
[4]   Columbus L. Roundup of Cloud Computing Forecasts and Market Estimates, 2014. Forbes Magazine. 2015 Jan.
[5]   El-korany AM, Khatab SM. Ontology-based Social Recommender System. *IAES International Journal of Artificial Intelligence*. 2012 Sep 1; 1(3): 127.

[6]   Adomavicius G and Tuzhilin A. (2005) Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17, 734-749. http://dx.doi.org/10.1109/TKDE.2005.99.

[7]   Porter MF. An algorithm for suffix stripping. *Program*. 1980 Mar 1; 14(3):130-7.

[8]   Manning C.D., Raghaven P and Schutze H. (2008) Introduction to Information Retrieval. Cambridge University Press, Cambridge. http://dx.doi.org/10.1017/CBO9780511809071

[9]   Sarwar B., Karypis G., Konstan J. and Riedl J. (2001) Item-Based Collaborative Filtering Recommendation. *Proceedings of the* 10*th International Conference on World Wide Web*, Hong Kong, 1-5 May 2001, 285-295. http://dx.doi.org/10.1145/371920.372071

[10] Claypool M., Gokhale A., Miranda T., Murnikov P., Netes D and Sartin M. (1999) Combining Content-Based and Collaborative Filters in an Online. Worcester Polytechnic Institute, Worcester

[11] Yang Y, Duan X. Cliques-based Data Smoothing Approach for Solving Data Sparsity in Collaborative Filtering. *Indonesian Journal of Electrical Engineering and Computer Science*. 2014 Aug 1; 12(8): 6324-31.

[12] Pham M.C., Cao Y., Klamma R. and Jarke M. (2011) A Clustering Approach for Collaborative Filtering Recommendation Using Social Network Analysis. *Universal Computer Science*, 17, 583-604.

BIOGRAPHIES OF AUTHORS

Mr. Siddharth Arun currently pursuing B.Tech - Computer Science and Engineering, in Amrita School of Engineering, Coimbatore. Areas of interest includes Cloud Computing, Big Data, and Automata Theory.

Mr. Aakash C. currently pursuing B.Tech - Computer Science and Engineering, in Amrita School of Engineering, Coimbatore. Areas of interest includes Data Structures and Algorithms.

Dr. Prakash P. currently serves as Assistant Professor at department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore. Areas of research include Cloud Computing, Automata Theory and Analysis of Algorithms.