Ontology-Based Indexing Method for Engineering Documents Retrieval

Weiguang Fang, Yu Guo, Wenhe Liao
College of Mechanical and Electrical Engineering
Nanjing University of Aeronautics and Astronautics
Nanjing 210016, China
e-mail: fang_weiguang@hotmail.com

***Abstract*—Engineering documents are valued resources in the reuse of engineering knowledge and effective reuse of these documents depends on efficient retrieval. A semantic indexing method, which accomplished by utilizing state-of-the-art ontology technologies of Semantic Web, is proposed in this paper to handle the issues in engineering documents retrieval. Firstly, in order to represent the semantics embedded in design documents, a domain ontology is constructed by concepts hierarchy and ontology population. Secondly, ontology inference service is presented to fulfill keywords semantic extension. Combining with Lucene mechanism, the extended document index is constructed for retrieval application. Finally, the classical matching and ranking approaches are adopted to develop a prototype domain knowledge retrieval system.**

Keywords-engineering documents; knowledge inference; ontology construction; semantic indexing

I. INTRODUCTION

Mechanical product design is in high demand for intelligence and creativity, which requires the designers to have a high level of skills and knowledge. Engineers have to obtain numerous design documents as references during the process of product design. Related studies show that approximately 70% of the engineers' working time is spent on searching for the corresponding design resources [1]. Besides, the procedure of designing new mechanical product is also a course of generating new design knowledge. For instance, more than 40,000 related documents are produced during the design of a single aero-engine in aerospace industry. However, with the explosion of engineering information, the majority of the knowledge contained in the documents is beyond any designer's grasp. Therefore, we have an expectation that we may develop a kind of retrieval system, in which engineering documents can be computer-understandable and assist designers to search for the exact same concepts as well as similar or related ones [2]. Fortunately, the Semantic Web technologies provide good solutions for us.

Currently, there are a variety of ways to query a semantic knowledge base and retrieve relevant results, which are classified into four categories, namely keyword-based, natural language-based, view-based and form based semantic querying [3]. Among them, keyword-based method is the most comfortable and easy-to-use in designers' eyes [4]. In keyword-based semantic querying, how to enhance the performance and scalability of retrieval system is our mainly concerned and ontology-based semantic indexing may be a good solution. So far, ontology-based information retrieval are discussed in many studies from all over the world. Kara et al. [5] proposed a semantic indexing method, enabling the users to query knowledge in an easy and convenient way. Chi[6] and Ameri et al. [7] took advantage of Semantic Web Rule Language (SWRL) to infer implicit knowledge in the domain ontology as well as the latent input query.

Our main contribution is a framework of ontology-based indexing for knowledge retrieval, which improves the knowledge extension issues in classical keyword-based method. Our proposed method utilize different semantic web technologies to construct inferred document index for knowledge retrieval purpose. This indexing mechanism has been successfully used for developing a domain knowledge retrieval system and observes improvements over tradition keyword-based retrieval system.

The rest of this paper is organized as follows: Section 2 gives the overall process of ontology-based indexing and semantic retrieval. In Section 3, the approach of ontology construction is illustrated. Three inference service for index construction are proposed in Section 4. The retrieval mechanism and ranking principle are illustrated in section 5. A brief conclusion this study is discussed in Section 6.

II. OVERALL PROCESS

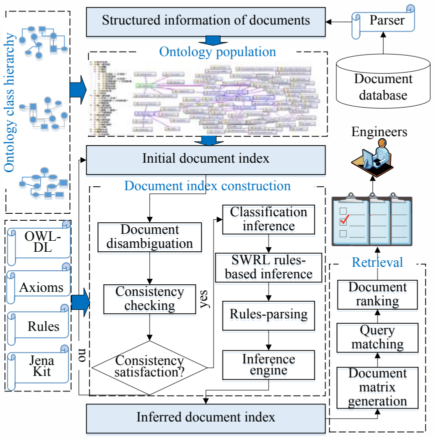


Figure 1. Overall process.

In order to take full advantage of semantics in design documents, we develop an ontology-based indexing approach for engineering documents retrieval. The overall process, shown in Figure 1, make the advantage of many cutting-edge

978-1-5090-3471-0/16/$31.00 ©2016 IEEE

technologies including OWL-DL, Axioms, rules, Jena kit and Lucene[8], aiming at building and extending documents indexes for the retrieval purpose.

It can be seen in Figure 1, the process of indexing start with parsing the unstructured information from engineering database into structured information. Four main aspects of the ontology-based indexing are illustrated as follows.

(a) Ontology class hierarchy. The domain taxonomy is utilized to provide a guidance for domain knowledge hierarchy. Also, this hierarchical model is specifically created as a reference for ontology population.

(b) Ontology population. This is a knowledge acquisition activity, which is a process of transforming or mapping the structured information of documents into ontology individuals. The second step is to add attributes of documents and other basic information related to engineering domain into individuals and properties of the ontology.

(c) Document index construction. This is the core process of our proposed approach and aims at construct inferred document index for retrieval application. By document disambiguation, the keywords in initial index are projected onto ontology tree. In addition, this part also provides all the necessary inference services including consistency checking, classification inference, and SWRL rules-based inference for keywords extension purpose. Lastly, combining with Lucene, we traverse the inference engine over OWL files to obtain the final inferred document index.

(d) Retrieval. The Vector Space Model (VSM) is employed to provide the mathematical foundation in our retrieval system. The detailed process and screenshot of our retrieval system can be seen in section 5.

III. Ontology construction

A. Ontology Class Hierarchy

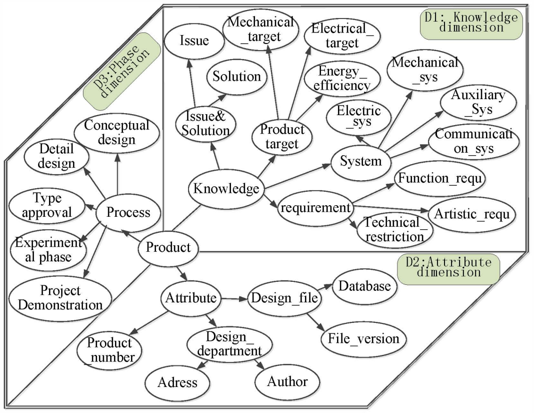


Figure 2. Ontology class hierarchy (radar domain).

The taxonomy provide an ideal backbone for engineering domain ontology construction, and a standard radar taxonomy is used to provide the hierarchy and seeds for radar ontology. The complex product development rely on multidisciplinary collaborative work, hence, it is hard to develop a common ontology to represent knowledge from different types of complex product. For example, the electromagnetism is a significant discipline in radar development. Comparing to it, the dynamic is one of the core disciplines in aircraft development, which is not necessary in radar domain. Therefore, the ontology in this paper only concentrate on the radar product domain, and the approach can also be extended to other product domains.

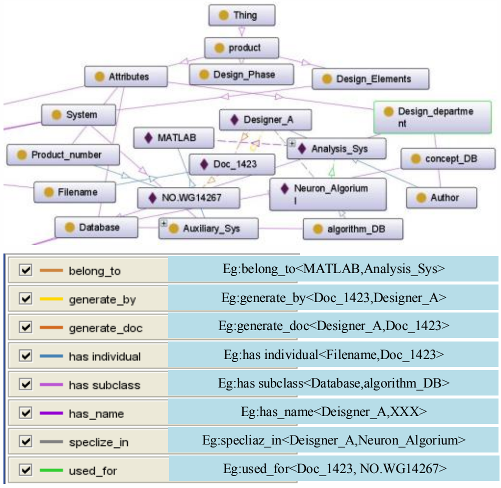
In the context of an engineering project, a group of engineers uses their specialized knowledges, following specific processes under well-defined topics. As such, the scope of radar ontology is able to be conveyed through three main dimensions: D1: knowledge dimension, D2: attribute dimension, and D3: process dimension [9]. The overall hierarchy of this ontology can be seen in Fig. 2.

The knowledge is divided by the radar domain taxonomy, which contains full domain concepts, generation/ specialization and composition/ aggregation relationship. Therefore, we impart the structure of taxonomy on radar ontology in order to facilitate users understanding and promote tenable integration.

In addition, the semantic relationship (e.g. Electrical performance is constrained by cable) between concepts is also indispensable to ontology construction and it provide the foundation for the ontology inference process. Under the guidance and collaborative works of specialists from different disciplines, the semantic relationship of radar ontology is defined and completed step by step.

B. Ontology Population

Ontology population aims at transforming or mapping the initial structured information of documents into ontology individuals and defining properties for them. Named entity recognizer is used for identifying the keywords in the documents. After that, we locate the positions of the outputting keywords in the ontology. Besides, the other attributes of documents such as ID, Filename, Author etc., are also added into ontology



☑ belong_to	Eg:belong_to<MATLAB,Analysis_Sys>
☑ generate_by	Eg:generate_by<Doc_1423,Designer_A>
☑ generate_doc	Eg:generate_doc<Designer_A,Doc_1423>
☑ has individual	Eg:has individual<Filename,Doc_1423>
☑ has subclass	Eg:has subclass<Database,algorithm_DB>
☑ has_name	Eg:has_name<Deisgner_A,XXX>
☑ speclize_in	Eg:specliaz_in<Deisgner_A,Neuron_Algorium>
☑ used_for	Eg:used_for<Doc_1423, NO.WG14267>

Figure 3. Example of ontology population.

Figure 3 shows the example of this process, the visualized figure is obtained from the plug-in ontoGraph of protégé 4.3

ontology editor. As a consequence, the domain ontology contains 277 domain-related concepts, 1637 individuals and 112 properties.

IV. Inferred Index Construction

The engineering ontology is constructed based on Web Ontology Language (OWL) and OWL-DL language in Section 3. More specifically, OWL-DL has function of Descriptions Logics (DLs) and it created to be computationally complete and decidable version of OWL language in order to pave the way for ontology inference. The overall process of ontology inference includes the following steps: A: document disambiguation; B: inferencing; C: index construction.

A. Document Disambiguation

Due to engineering documents have features such as syntax variations and semantic complexities, a proper document disambiguation is imperative during index construction [10]. After experiencing the document preprocessing, documents are divided up into a set of keywords.

Firstly, searching for corresponding individuals in ontology for each extracted keyword in document. Through matching between keyword in the document and synonyms of individual, $Iscore$ is calculated to measure relevance between keyword and individual. Supposing that $D_k = \{K_1, K_2, \cdots, K_l\}$ represents a set of keywords in document, $I = \{I_1, I_2, \cdots, I_m\}$ represents all the individuals in ontology, and $L(I_j) = \{L_{j1}, L_{j2}, \cdots, L_{jn}\}$ indicates the set of synonyms of the individual j. The $Iscore$ is calculated by the equation (1):

$$Iscore(D_i, I_j) = \max\left(\frac{number\ of\ synonyms\ in\ L(I_j)\ matched\ with\ K_i}{number\ of\ keywords\ in\ L_{jk}}\right) \quad (1)$$

In addition, a set of individuals is named as $CI(T_i)$. When $CI(T_i) = \{I_j \mid Iscore(D_i, I_j) \geq \beta\}$, it is means that a set of individuals have a bigger $Iscore$ than threshold β. Hence, this set can be specified as corresponding individuals for the keyword i and an individual which has the maximum $Iscore$ is selected as the disambiguated meaning of the keyword.

B. Inference Service

The inference service in our system is designed for extending the keywords in indexes. This module mainly contains three parts: (a) consistency checking, (b) classification inference, and (c) SWRL rules-based inference.

(a) Consistency checking. This part aims at ensuring that there is no contradictory assertion in radar ontology. We define a series of property restrictions during ontology construction, and they could be divided into two types: value constraints, and cardinality constraints. The first constraint type, for example, to confine that only red cable (subclass of cable in ontology) is allowed to transmit high tension electricity, and utilizing cardinality could confine that only one purple cable is allowed in radar cabinet. Furthermore, these restrictions also could be used for inferring new information, e.g. if the value of a property whose range is confined to a class, then the type of its individual could be inferred.

(b) Classification inference. According to the class-subclass and class-instance definitions in radar ontology, we can acquire all the necessary classes and instances by classification inference. In this paper, we define the model of two types of classification inferences as:

$$< C_s, I_s > \begin{cases} \xrightarrow[rel]{forward()} For(< C_s^*, I_s^* >, rel) \\ \xrightarrow[rel]{reverse()} Re(< C_s^*, I_s^* >, rel) \end{cases} \quad (2)$$

where $< C_S, I_S >$ is used to represent the set of Class and Instance for inferring, $forward()$ and $reverse()$ respectively represent forward inference and reverse inference. rel is the relation between ontology nodes, $< C_S^*, I_S^* >$ represents the result of inference. An application example can be seen in Fig. 1 (the orange parts).

(c) SWRL rules-based inference. In order to discover more information by semantic relationship defined in ontology, SWRL rules-based inference is used for implicit knowledge mining. SWRL is a proposed language for the Semantic Web that can be used for expressing rules and logic[11]. To illustrate the ability of SWRL rules-based inference, we list two examples of SWRL rules as:

Rule5:

$$[microwaveIntegratedCircuit\ \ ?x$$
$$?x\ hasModule\ \ ?y$$
$$?y\ needSoftware\ \ ?z \rightarrow ?x\ needSoftware\ \ ?z\ ]$$

Rule5 means that if a Microwave Integrated Circuit (MIC) x has module y and y need software z, so, software z can be inferred through these SWRL rules.

Rule6:

$$[lowNoiseAmplifier\ \ ?x$$
$$?x\ subclassOf\ \ ?y$$
$$?y\ useCircuitType\ \ ?z \rightarrow ?x\ useCircuitType\ \ ?z\ ]$$

Rule6 means that if a Low Noise Amplifier (LNA) x is subclass of y and y use circuit type z, so circuit type z can be inferred.

Combining with above mentioned classification inference, an example (Figure. 4) is given to illustrate the process of overall knowledge extension by inference.

It can be seen in Figure 4 that the keyword ‘Microwave integrated circuit’ can be extended to a series of keywords in knowledgebase. And this process is accomplished by reverse direction inference, forward direction inference, and SWRL rules-based inference.

Currently, Java Expert System Shell (JESS) is a kind of inference engine which is most commonly used for SWRL rules reasoning. However, JESS can only be embedded in protégé to implement inference and is hardly integrated into system development. According to this, we propose a solution that to separate ontology file and SWRL rules from inference engine, storing OWL (ontology file) and XML (SWRL rules) separately. The overall process of inference is executed by Jena development kit, which parse the ontology file and SWRL rules and transform the rules into recognizable version. This main algorithm of inference engine can be seen in Figure. 5.

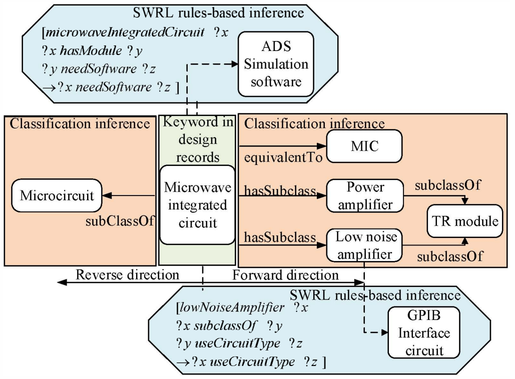


Figure 4. Example of inference service in index construction.

```
Input: The radar ontology as defined above(pd_retrieval.owl), SWRL
rules(swrl_rules.xml)
Output: A set of inferred relations between concepts
Algorithm: Inference engine
   InferenceRelation(Resource a, Resource b){
            Model model_Inf = ModelFactory.creatDefaultModel();
            model_Inf.read("file:D:/KM/ontology/
       pd_retrieval.owl");//parse OWL file
            List swrl_rules = Rule.rulesFromURL("file:D:/KM/
   ontology/swrl_rules.xml");//parse and transform SWRL rules
            GenericRuleReasoner reasoner = new
   GenericRuleReasoner(swrl_rules);//construct inference engine
            reasoner.setOWLTranslation(true);
            reasoner.setDerivationLogging(true);
            reasoner.setTransitiveClosureCaching(true);
            OntModel ont_model = ModelFactory.creatOntologyModel
            (OntModelSpec.OWL_MEM_RULE_INF, model_Inf);
            Resource config = om.creatResource;
            config.addProperty(ReasonerVocabulary.PROPruleMode,"
   hybrid");
            InfModel inf_model =
   ModelFactory.creatInfModel(reasoner,ont_model);
            StmtIterator stmt_iter =
   inf_model.listStatements(a,null,b);//Infer implicit relations
   between a and b}
```

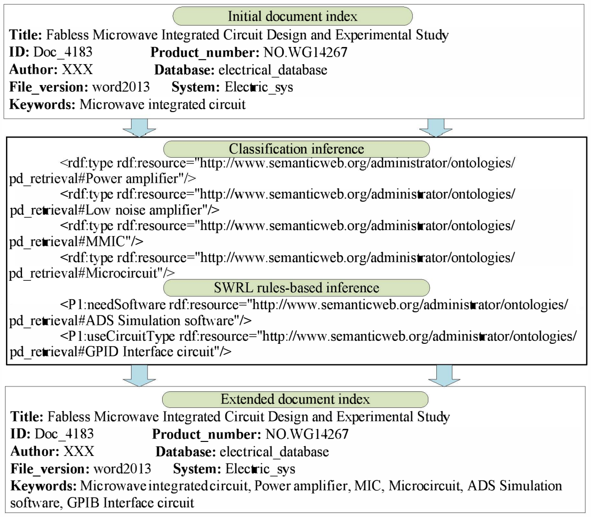
Figure 5. The main algorithm of inference engine.

C. Index Construction

The index construction is one of the most important procedure in developing a retrieval system. The Lucene mechanism is employed to build an initial document index and include the basic information of document (e.g. Title, ID, Author, etc.). Moreover, the keyword in initial index is projected onto radar ontology by document disambiguation, then the inference services could be implemented to extend the keyword. For the OWL files after inference, we construct an extended version of the initial index. In addition to the extended keywords are included in the extended document index, the basic information from initial index is also inherited. The simplified index structure and parts of inferred OWL file can be seen in Fig. 5.

V. Semantic Retrieval

After constructing the document index, the keywords in initial index are extended to a set of inferred keywords. Next, the Vector Space Model (VSM) could be employed to offer the mathematical foundation in our semantic retrieval system[12]. The retrieval section can be divided into three parts: A: semantic matrix generation, B: matching, and C: ranking. The screenshot of our retrieval system is also given in part D.



Initial document index

Title: Fabless Microwave Integrated Circuit Design and Experimental Study
ID: Doc_4183 **Product_number:** NO.WG14267
Author: XXX **Database:** electrical_database
File_version: word2013 **System:** Electric_sys
Keywords: Microwave integrated circuit

Classification inference

```
        <rdf:type rdf:resource="http://www.semanticweb.org/administrator/ontologies/
pd_retrieval#Power amplifier"/>
        <rdf:type rdf:resource="http://www.semanticweb.org/administrator/ontologies/
pd_retrieval#Low noise amplifier"/>
        <rdf:type rdf:resource="http://www.semanticweb.org/administrator/ontologies/
pd_retrieval#MMIC"/>
        <rdf:type rdf:resource="http://www.semanticweb.org/administrator/ontologies/
pd_retrieval#Microcircuit"/>
```

SWRL rules-based inference

```
        <P1:needSoftware rdf:resource="http://www.semanticweb.org/administrator/ontologies/
pd_retrieval#ADS Simulation software"/>
        <P1:useCircuitType rdf:resource="http://www.semanticweb.org/administrator/ontologies/
pd_retrieval#GPID Interface circuit"/>
```

Extended document index

Title: Fabless Microwave Integrated Circuit Design and Experimental Study
ID: Doc_4183 **Product_number:** NO.WG14267
Author: XXX **Database:** electrical_database
File_version: word2013 **System:** Electric_sys
Keywords: Microwave integrated circuit, Power amplifier, MIC, Microcircuit, ADS Simulation software, GPIB Interface circuit

Figure 6. Inferred OWL file and index structure.

A. Semantic Matrix Generation

For the purpose of weighting the initial and inferred keywords, the classical TF-IDF model is used to measure the weights of keywords. Then, the document could be represented as vector $d_i = \{w_{1,i}, w_{2,i}, w_{3,i}, \cdots, w_{m,i}\}$, where $w_{m,i}$ denotes the weights of the keyword m in document i . It is noted that the keywords in vector also contains the inferred keywords.

Therefore, the structure of our documents semantic matrix is as follows:

$$W(w_{i,j})_{M\times N} = \left[\begin{array}{cccccc} & d_1 & d_2 & d_3 & \cdots & d_n \\ c_1 & w_{1,1} & w_{1,2} & w_{1,3} & \cdots & w_{1,n} \\ c_2 & w_{2,1} & w_{2,2} & w_{2,3} & \cdots & w_{2,n} \\ c_3 & w_{3,1} & w_{3,2} & w_{3,3} & \cdots & w_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ c_m & w_{m,1} & w_{m,2} & w_{m,3} & \cdots & w_{m,n} \end{array}\right] \tag{3}$$

When the user input the query, the keywords in the query are able to be regarded as a document vector q_{n+1} . Therefore, the same procedure is also applied to this new document (query) to the calculate the values of each keyword in q_{n+1} , which is added as a column, forming a query-document matrix $QW(w_{i,j})_{M\times(N+1)}$.

B. Matching

Given that the inferred documents indexes and query are represented in the same query-document matrix

$QW(w_{i,j})_{M\times(N+1)}$, the matching process is to compare the query vector q_{n+1}, against each document vector d_i using a measure cosine similarity.

$$\cos(\overrightarrow{\mathrm{d}_i}, \overrightarrow{q_{n+1}}) = \frac{\overrightarrow{\mathrm{d}_i} \bullet \overrightarrow{q_{n+1}}}{\left\|\overrightarrow{\mathrm{d}_i}\right\| \left\|\overrightarrow{q_{n+1}}\right\|} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n+1} \mathrm{d}_i \bullet q_{n+1}}{\sqrt{\sum_{i=1}^{n} \mathrm{d}_i^{\,2}} \bullet \sqrt{\sum_{i=1}^{n} q_{n+1}^{\,2}}} \quad (4)$$

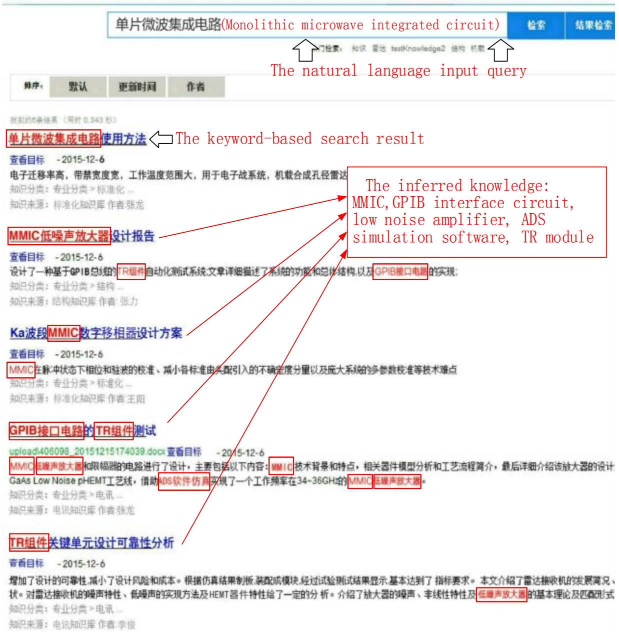
The cosine similarity measure can also be thought as the angle that separates the vectors, and the value of $\cos(\overrightarrow{\mathrm{d}_i}, \overrightarrow{q_{n+1}})$ is between $(0,1)$.

C. Ranking

Our ranking principles are also based on the mature Lucene mechanism, and its default ranking principle will give good results on the whole. However, in order to fully take the advantages of our ontology-based indexing method, the ranking principle of Lucene is slighted modified by us. The overall ranking principles can be summarized as follows:

- The documents which contain keywords exact matching query rank ahead of others.
- The documents which contain more inferred information rank ahead of the others contain less.
- The documents which contain inferred information rank ahead of the other documents include none.

D. Screenshot of Retrieval System


检索
结果检索
The natural language input query
The keyword-based search result
The inferred knowledge:
MMIC, GPIB interface circuit,
low noise amplifier, ADS
simulation software, TR module

单片微波集成电路(Monolithic microwave integrated circuit)

热门检索： 知识 雷达 testKnowledge2 结构 机载

排序： 默认 更新时间 作者

找到约6条结果（用时 0.343 秒）

单片微波集成电路使用方法

查看目标 -2015-12-6

电子迁移率高，带禁宽度宽，工作温度范围大，用于电子战系统，机载合成孔径雷达

知识分类：专业分类 > 标准化 ...

知识来源：标准化知识库 作者:张龙

MMIC低噪声放大器设计报告

查看目标 -2015-12-6

设计了一种基于GPIB总线的TR组件自动化测试系统.文章详细描述了系统的功能和总体结构.以及GPIB接口电路的实现;

知识分类：专业分类 > 结构 ...

知识来源：结构知识库 作者: 张力

Ka波段MMIC数字移相器设计方案

查看目标 -2015-12-6

MMIC在脉冲状态下相位和驻波的校准、减小各标准由失配引入的不确定度分量以及庞大系统的多参数校准等技术难点

知识分类：专业分类 > 标准化 ...

知识来源：标准化知识库 作者:王阳

GPIB接口电路的TR组件测试

upload\406098_20151215174039.docx 查看目标 -2015-12-6

MMIC低噪声放大器和限幅器的电路进行了设计，主要包括以下内容：MMIC技术背景和特点，相关器件模型分析和工艺流程简介，最后详细介绍该放大器的设计

GaAs Low Noise pHEMT工艺线，借助ADS软件仿真实现了一个工作频率在34~36GHz的MMIC低噪声放大器。

知识分类：专业分类 > 电讯 ...

知识来源：电讯知识库 作者:张龙

TR组件关键单元设计可靠性分析

查看目标 -2015-12-6

增加了设计的可靠性.减小了设计风险和成本。根据仿真结果制板.装配成模块.经过试验测试结果显示.基本达到了 指标要求。本文介绍了雷达接收机的发展简况、现状。对雷达接收机的噪声特性、低噪声的实现方法及HEMT器件特性给了一定的分 析。介绍了放大器的噪声、非线性特性及低噪声放大器的基本理论及匹配形式

知识分类：专业分类 > 电讯 ...

知识来源：电讯知识库 作者:李俊

Figure 7. Screenshot of retrieval system in radar domain.

The proposed ontology-based indexing approach has been implemented in a keyword-based retrieval system of radar design knowledge management platform (Fig. 7). It is worth to note that our retrieval system is implemented in a Chinese radar design and manufacturing enterprise, so the input and output of system are Chinese-based. As is shown in Figure 8, we also type 'microwave integrated circuit' as input query, the top text box shows the natural language input query and the bottom panel shows the retrieval results. It can be seen in Figure 8 that the record ranking first is the only outputting document of keyword-based search, and there are no other exact matching results. However, with the help of ontology-based indexing, other inferred knowledge such as 'MMIC', 'low noise amplifier', 'TR module', etc., are also retrieved by our system. This is mainly attributed to that the keyword in initial document has been extended during ontology inference. Hence, implementing ontology-based indexing is a useful and convenient approach to enhance the recall ratio of retrieval system.

VI. Conclusion

A novel semantic retrieval framework is presented in this paper and the proposed method has been applied in radar knowledge retrieval system. The framework contains many aspects of Semantic Web, namely, ontology construction, ontology population, ontology inference, semantic indexing, and semantic retrieval.

Acknowledgment

This work is supported by National Natural Science Foundation of China [grant number 51175287].

References

[1] Z. Li, V. Raskin, and K. Ramani, "Developing engineering ontology for information retrieval," Journal of Computing & Information Science in Engineering, vol. 8, pp. 504-505, 2008.

[2] S. Ma and L. Tian, "Ontology-based semantic retrieval for mechanical design knowledge," International Journal of Computer Integrated Manufacturing, vol. 28, pp. 226-238, 2015.

[3] X. Zhang, X. Hou, X. Chen, and T. Zhuang, "Ontology-based semantic retrieval for engineering domain knowledge," Neurocomputing, vol. 116, pp. 382-391, 2013.

[4] L. Li, F. Qin, S. Gao, and Y. Liu, "An approach for design rationale retrieval using ontology-aided indexing," Journal of Engineering Design, vol. 25, pp. 259-279, 2013.

[5] S. Kara, Ö. Alan, O. Sabuncu, S. Akpınar, N. K. Cicekli, and F. N. Alpaslan, "An ontology-based retrieval system using semantic indexing," Information Systems, vol. 37, pp. 197-202, 2010.

[6] Y. L. Chi, "Rule-based ontological knowledge base for monitoring partners across supply networks," Expert Systems with Applications, vol. 37, pp. 1400-1407, 2010.

[7] F. Ameri and C. Mcarthur, "Semantic rule modelling for intelligent supplier discovery," International Journal of Computer Integrated Manufacturing, vol. 27, pp. 570-590, 2014.

[8] Apache Lucene. In the Apache Software Foundation. Retrieved May 10, 2016. Available: https://lucene.apache.org/

[9] W. Fang, Y. Guo, W Liao, F Wang, "The knowledge representation and annotation method based on ontology for complex products design," Computer Integrated Manufacturing Systems, to be published. Available: http://www.cnki.net/kcms/detail/11.3619.TP.20160530.0944.012.html

[10] G. J. Hahm, J. H. Lee, and H. W. Suh, "Semantic relation based personalized ranking approach for engineering document retrieval," Advanced Engineering Informatics, vol. 29, pp. 366-379, 2015.

[11] SWRL. In World Wide Web Consortium (W3C). Retrieved May 10, 2016. Available: https://www.w3.org/Submission/SWRL/

[12] G. Salton, A. Wong, and C. S. Yang, A vector space model for automatic indexing: Morgan Kaufmann Publishers Inc., 1997.