تطبيق أسلوب التخزين المزدوج لحل مشكلة القيم الخالية والسمات متعددة القيم في تخزين الويب الدلالي

أشرف فاروق أبوبكر محمود و الطيب السماني عبدالجبار و أسامة محمد عبدالله يوسف

Al Neelain University

GCNU Journal

ISSN: 1858-6228

Volume 16 - 2021
Issue: 03

Graduate College
Al Neelain University

Graduate College Journal – NU
Vol.16 -2021, No. (3)
ISSN: 1858-6228, http://www.neelain .edu.sd

مجلة الدراسات العليا - جامعة النيلين
المجلد (16) –2021، العدد (3)
الرقم الدولي الموحد للدوريات: 1858-6228

تطبيق أسلوب التخزين المزدوج لحل مشكلة القيم الخالية والسمات متعددة القيم في تخزين الويب الدلالي

أشرف فاروق أبوبكر محمود[1]، الطيب السماني عبدالجبار[2]، أسامة محمد عبدالله يوسف[1]

[1]عمادة السنة التحضيرية والدراسات المساندة، جامعة الحدود الشمالية، عرعر-المملكة العربية السعودية

[2]كلية علوم الحاسوب و تقانة المعلومات، جامعة النيلين، الخرطوم-السودان ،

المستخلص

تهدف هذه الورقة إلى توفير حل مشكلة التخزين في الويب الدلالي الذي لديه بعض القيود من حيث الكفاءة والتدرجية بسبب طبيعة إطار وصف الموارد RDF/XMLبناءً على التنسيق الثلاثي (الموضوع، الخاصية، الكائن) و الذي يؤدي إلى ظهور مشكلة القيم الخالية في جدول الخصائص(nulls) ، والسمات متعددة القيم (Replication) .، و الحل المقدم هنا هو أسلوب التقسيم العمودي الذي يتم فيه استخدام عمودين لكل خاصية، ومن ناحية أخرى من خلال إلقاء نظرة على تقنية جدول الخصائص من حيث المفهوم والايجابيات والسلبيات. علاوة على ذلك، يتم تقديم وصف مختصر حول نهج تخزين الأعمدة.

الكلمات المفتاحية: الويب الدلالي، Resource Description Framework (RDF)،XML ، تقنية جدول الخصائص، نهج التقسيم العمودي

المقدمة

محرك بحث الويب هو برنامج مصمم للمساعدة في العثور على المعلومات المخزنة على الشبكة العنكبوتية العالمية. ويسمح محرك البحث لأي مستخدم بالسؤال عن محتوى يفي بمعايير محددة (عادةً تلك التي تحتوي على كلمة أو عبارة معينة) ويسترجع قائمة من المراجع التي تطابق تلك المعايير. فمحركات البحث هي في الأساس فهارس النص الكامل الضخمة لصفحات الويب ويستخدمون فهارس يتم تحديثها بانتظام للعمل بسرعة وكفاءة. إن جودة الفهارس وكيفية استخدام المحركات للمعلومات التي تحتويها هي التي ترفع - أو تحط – من جودة نتائج البحث. فبالرغم من أن فهارس محرك البحث متشابهة، لكنها أكثر تعقيدًا من فهارس الكتب. ومعرفة بعض الشيء عن كيفية بنائها واستخدامها يمكن أن يحسن إلى حد كبير من مهارات البحث. هناك أنواع أخرى من محركات البحث وهي محركات البحث في المؤسسات والتي تبحث في الإنترانت ومحركات البحث الشخصية والتي تبحث في أجهزة الحاسوب الشخصية الفردية ومحركات البحث بالهاتف المحمول. تستخرج بعض محركات البحث أيضًا البيانات المتوفرة في مجموعات الأخبار أو قواعد البيانات الكبيرة أو الدلائل المفتوحة مثل. DMOZ.org. فعلى عكس أدلة الويب التي يتحكم بها محررون بشريين تعمل محركات البحث خوارزميًا.

كان أول محرك بحث على الويب هو 'Wandex' تم جمعه بواسطة World Wide Web Wanderer والزاحف تم تطويره بواسطة Matthew Gray في معهد ماساتشوستس للتكنولوجيا في عام 1993. وهناك محرك بحث Aliweb وقد ظهر مبكرا جدًا في عام 1993 ولا يزال يعمل حتى اليوم. وكان أول محرك بحث يعتمد على 'النص الكامل' هو WebCrawler الذي صدر عام 1994. والذي على عكس سابقيه فإنه يتيح للمستخدمين البحث عن أي كلمة في أي صفحة ويب والتي أصبحت المعيار لجميع محركات البحث الرئيسية منذ ذلك الحين. وفي نفس العام 1994 ظهر أيضا Lycosالذي بدأ في جامعة كارنجي ميلون واكتسب صيتا تجاريا واسعا. بعد فترة وجيزة ظهرت العديد من محركات البحث وتنافسوا على شعبيتها. وشملت (Excite، Infoseek ، Inktomi، Northern Light وAltaVista) و تنافسوا مع الأدلة المشهورة في بعض النواحي مثل Yahoo. !. كما تم دمج الأدلة أو إضافتها فيما بعد في تقنية محرك البحث للحصول على وظائف أكبر.

يسمى الويب التقليدي (2Web) بمطابقة الكلمات وتم تطويره إلى الويب الدلالي(Web3)، وهذا يعني البحث بمعنى الكلمة.

الويب الدلالي هو امتداد للويب الحالي [1] والذي يسمح بوصف معنى المعلومات بدقة من حيث المفردات المحددة جيدًا والتي يفهمها الأشخاص وأجهزة الكمبيوتر. ويتم وصف معلومات الويب الدلالي باستخدام معيار W3C) اتحاد شبكة الويب العالمية) ويسمى إطار وصف الموارد (RDF).

البحث الدلالي على الويب هو محرك بحث للويب الدلالي. ويمكن استخدام مواقع الويب الحالية من قِبل كل من الأشخاص وأجهزة الكمبيوتر لتحديد موقع المعلومات المنشورة على الويب الدلالي بدقة وجمعها [2]. تعد خرائط المفاهيم أحد أهم المفاهيم المستخدمة في البنية التحتية لموقع الويب الدلالي وRDFS (إطار وصف / مخطط الموارد) وOWL (لغة أنطولوجيا الويب) نموذجان لتمثيل البيانات الموصي بهما من W3C لتمثيل خرائط المفاهيم. سيدعم الويب الدلالي الاكتشاف التلقائي وتكامل البيانات وإعادة استخدامها بطريقة أكثر فاعلية وتوفير الدعم لمشكلة التشغيل البيني التي لا يمكن حلها باستخدام تقنيات الويب الحالية.

يمكن للويب الدلالي [3] حل المشكلة الأولى في الويب باستخدام التعليقات التوضيحية الدلالية لإنتاج معلومات ذكية وذات معنى باستخدام آلية واجهة الاستعلام وخرائط المفاهيم. يمكن حل مشكلة أخرى بواسطة نماذج الاستعلام المستندة إلى الرسم البياني [4]. تتطلب الشبكة الدلالية حل المشكلات الصعبة للغاية في مجالات تمثيل المعرفة وفهم اللغة الطبيعية.

أهمية الورقة

بناء على ما ذكر في المقدمة يتضح أن هناك بعض العيوب التي تظهر عند إستخدام الويب الدلالي و(القيم الخالية و التكرار) و المرتبطة بالتخزين في الويب الدلالي الذي لديه بعض القيود من حيث الكفاءة والتدرجية بسبب طبيعة إطار وصف الموارد RDF بناءً على التنسيق الثلاثي (الموضوع، الخاصية، الكائن).

اهداف الورقة

I. حل مشكلة القيم الخالية في جدول الخصائص.
II. حل مشكلة السمات متعددة القيم.

منهج الورقة في معالجة المشكلة

الحل المقدم هنا هو أسلوب التقسيم العمودي الذي يتم فيه استخدام عمودين لكل خاصية. كما هو موضح في المثال التالي.

نهج التقسيم العمودي

يتم فرز كل جدول حسب الفاعل بحيث يمكن تحديد مجموعة الفاعل المعينة بسرعة، ويمكن استخدام روابط الجمع السريعة هذه لإعادة إنشاء معلومات حول خصائص متعددة للمجموعات الفرعية من الفاعل. ويمكن أيضًا فهرسة عمود القيمة لكل جدول.

جدول (1) التقسيم العمودي للفئة (**isNamed**)

Subject	isNamed
NO1	"Ahmed Omer"
NO2	"Ali Mohammed"
NO3	"John Krees"
NO4	"James Kan"

يستخدم الجدول (1) أسلوب التقسيم العمودي. ويحتوي على عمودين، الفاعل الذي يتضمن جميع أرقام الموظفين والخاصية 'isName' التي تتضمن جميع أسماء الموظفين.

جدول (2) التقسيم العمودي للفئة (**hasEmployee**)

Subject	hasEmployee
NO1	person1
NO2	person2
NO3	person3
NO4	Person4

يستخدم الجدول (2) أسلوب التقسيم العمودي. ويحتوي على عمودين، الفاعل الذي يتضمن جميع أرقام الموظفين والخصائص 'hasEmployee' والتي تتضمن وصف جميع الموظفين.

جدول (3) التقسيم العمودي للفئة (**isCalled**)

Subject	isCalled
NO1	"Blue Nile"
NO2	"Blue Nile"
NO3	"Blue Nile"
NO4	"Blue Nile"

يستخدم الجدول (3) أسلوب التقسيم العمودي. ويحتوي على عمودين، الأول هو الفاعل الذي يشمل جميع أرقام الموظفين والثاني هو الخاصية 'isCalled' التي تتضمن اسم الشركة.

جدول (4) التقسيم العمودي للفئة (**hasAddress**)

Subject	has Address
NO1	"KSA"
NO3	"Canada"
NO4	"London"

يستخدم الجدول (4) أسلوب التقسيم العمودي. ويحتوي على عمودين، الأول هو الفاعل والذي يشمل جميع أرقام الموظفين التي لديها عنوان والثاني هو الخاصية 'hasAddress' والتي تشمل عناوين الموظفين.

جدول (5) التقسيم العمودي للفئة (**hasPhone**)

Subject	hasPhone
NO1	0147895478
NO2	0145258147
NO4	0547896214

يستخدم الجدول (5) أسلوب التقسيم العمودي. ويحتوي على عمودين، الأول هو الفاعل والذي يشمل جميع أرقام الموظفين التي لديها عنوان والثاني هو الخاصية 'hasPhone' والتي تشمل ارقام الاتصال للموظفين.

مزايا نهج التقسيم العمودي

هذا النهج له مزايا متعددة تشمل دعم السمة متعددة القيم - وبالتالي يتم تمثيل الفاعلين متعددي القيم كصفوف متعددة في الجدول مع نفس الفاعل وقيم مختلفة (انظر الجدول (6))، ودعم السجلات غير المتجانسة – الفاعلين (على سبيل المثال 15NO) التي لا تحدد خاصية معينة (hasAddress / hasPhone) يتم حذفها ببساطة من الجدول لتلك الخاصية (انظر الجدولين 4 و5)، فقط تلك الخصائص التي تم الوصول إليها بواسطة استعلام تحتاج إلى قراءة. وسيتم تقليل تكاليف الإدخال / الإخراج بشكل جيد، ولا توجد حاجة إلى خوارزميات التجميع. وخوارزمية إنشاء الجداول في أسلوب التقسيم العمودي واضحة ولا تحتاج إلى أي تغييرات بمرور الوقت ولديها عدد أقل من الاتحادات (نسبية إلى نهج مخطط فئة الخاصية). كما توجد جميع البيانات الخاصة بالخاصية محددة في نفس الجدول، فقواعد الاتحاد في الاستعلامات أقل شيوعًا [5].

جدول (6) التقسيم العمودي متعدد القيم

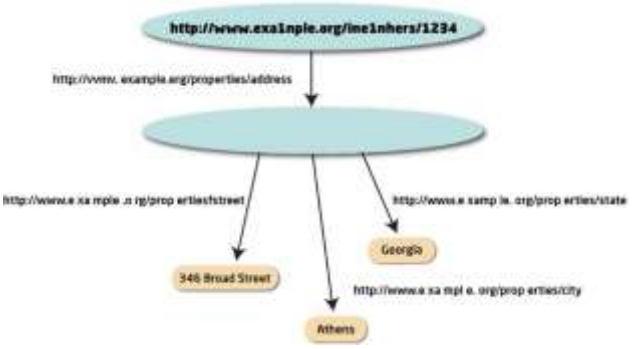
Subject	hasEmployee
NO15	"Ahmed Omer"
NO15	"Ali Mohammed"
NO15	"John Krees"
NO15	"James Kan"

يوضح الجدول (6) كيف يتم تخزين السمة متعددة القيم باستخدام تقنية التقسيم العمودي. ويتكون من عمودين، أولاً، رقم معرّف الشركة، وثانيًا، أسماء الموظفين. ويعرض العلاقة بين الشركة والموظفين (من واحد إلى أكثر).

النتائج والمناقشة

إطار وصف الموارد RDF: Resource Description Framework

إطار وصف المصدر هو الإطار العام لكيفية وصف الموارد على الإنترنت مثل الموقع ومحتواه. ويطلق عليه "الميتا ديتا" أو بيانات حول البيانات.



الشكل (1) الميتا ديتا (**Meta Data**)

إطار وصف المورد هو معيار لوصف الموارد على الويب الدلالي. ويمثل البيانات كتعابير يكون مصدرها تقريبا هو استخدام ربط الشكل (2) وعُقد الموارد وقيم خصائصها مع الأقواس المصنفة التي تمثل الخصائص. فيما يخص بناء الجملة قد يكون الشكل اعلاه ممثلا باستخدام XML بناء جملة (RDF/XML). وفي الواقع هو تنسيق تبادل بيانات RDF ومع ذلك قد يتم تحليل الشكل (1) هيكليا في سلسلة من الثلاثيات (subjectالفاعل، propertyالخاصية، object المفعول به). يمكن أن تكون تلك الثلاثيات المخزنة في قاعدة بيانات علائقية مع مخطط ثلاثة أعمدة [6].

تستخدم معرفات الويب (URIs) لتمييز الموارد وتصف الموارد مع الخصائص والقيم التي تكون الخاصية عبارة عن مورد له اسم (مثل الموظف) وقيمة الخاصية هي "أحمد عمر".

على سبيل المثال لتمثيل حقيقة أن (أحمد عمر وعلي محمد وجون كريس وجيمس كان) موظفون في شركة تدعى "النيل الأزرق"، سيتم استخدام خمسة عشر ثلاثية

Person1 isNamed "Ahmed Omer"
Person2 isNamed "Ali Mohammed"
Person3 isNamed "John Krees"
Person4 isNamed "James Kan"
Person1 hasAdress "KSA"
Person3 hasAddress "Canada"
Person4 hasAddress "London"
Person1 hasPhone "0147895478"
Person2 hasPhone "0145258147"
Person4 hasPhone "0547896214"
Company hasEmployee person1
Company hasEmployee person2
Company hasEmployee person3
Company hasEmployee person4

Company isCalled "Blue Nile

جدول (7) مثال للعناصر الثلاثية في**RDF**

Subject	Property	Object
NO1	isNamed	"Ahmed Omer"
NO2	isNamed	"Ali Mohammed"
NO3	isNamed	"John Krees"
NO4	isNamed	"James Kan"
NO5	hasEmployee	Person1
NO6	hasEmployee	Person2
NO7	hasEmployee	Person3
NO8	hasEmployee	Person4
NO9	has Address	Person1
NO10	has Address	Person3
NO11	has Address	Person4
NO12	hasPhone	Person1
NO13	hasPhone	Person2
NO14	hasPhone	Person4
NO15	isCalled	"Blue Nile"

يمثل الجدول أعلاه مفهوم تخزين الثلاثيات في الويب الدلالي. ويتكون من ثلاثة أعمدة على النحو التالي: الفاعل، الخاصية، المفعول به. ويحتوي عمود الفاعل على أرقام الأشخاص، أما عمود الخاصية هو وصف للفاعل، ويتضمن عمود المفعول به قيم الخصائص.

التخزين في الويب الدلالي

قواعد بيانات RDF عبارة عن مجموعات من 'ثلاثيات المعرفة'، حيث يكون كل ثلاثة من النموذج (الفاعل، الفعل، المفعول به) ونماذج العلاقة الثنائية للفعل بين الفاعل والمفعول به. ويمكن أن ينظر إليه على أنه رسم بياني موجه. يجب أن يدعم نظام تخزين RDF الفعال تقييمًا سريعًا لأنماط الرسم البياني ومدى قواعد بيانات RDF التي تحتوي على ملايين من الثلاثيات. ولا يبدو التنفيذ العلائقي المباشر مواتٍ للغاية، وتحديداً علاقة ثلاثية مع موضوع ثلاثة أعمدة فاعل، فعل، ومفعول به يحمل كل ثلاثيات RDF: فالمشكلة الأساسية في هذا النهج هي أن تقدير أنماط الرسوم البيانية التي تم جمعها يتطلب عادةً كمية كبيرة من الارتباط الذاتي المكلف في هذا الجدول. وتتمثل الفكرة الأقل تعقيدًا من الناحية النظرية في إعداد جدول واحد لكل تقييم محدد في البيانات، والذي يمكن اعتباره تقسيمًا رأسيًا كاملاً على المستندات. ويشتمل كل فعل على أعمدة (فاعل، ومفعول به) ويتضمن جميع أزواج الفاعل، والمفعول به المتصلة من خلال الفعل الخاص. ثم يتم توزيع البيانات عبر عدة جداول أصغر وعندما يتم تسوية الفعل، لا تتضمن الروابط مجموعة الثلاثيات كاملة. فمن خلال فرز البيانات فعليًا في عمود الفاعل، قد يتم التعرف على روابط الفاعل بين جدولين، وهي عملية متكررة، في وقت خطي من خلال الجمع بين أعمدة الفاعل [7].

تقنية جدول الخصائص

يستخدم جدول الخصائص لزيادة كفاءة استرجاع المعلومات من المخازن الثلاثية. هناك نوعان من جداول الخصائص. حيث يحتوي النوع الأول، الذي يسمى جدول الخصائص متفاوت المسافات، على مجموعات من الخصائص التي لها ميل لوصفها معا. ويعرض الثاني نوع جدول الخصائص، والذي يطلق عليه جدول فئة الخصائص، والذي يستغل نوع خصائص الفاعل لتجميع مجموعات من الفاعل قابلة للمقارنة معًا في نفس الجدول [8].

جدول (8) تجميع الخصائص

Subject	isNamed	hasEmployee	isCalled	has Address	hasPhone
NO1	"Ahmed Omer"	person1	"Blue Nile"	"KSA"	0147895478
NO2	"Ali Mohammed"	person2	"Blue Nile"	NULL	0145258147
NO3	"John Krees"	Person3	"Blue Nile"	"Canada"	NULL
NO4	"James Kan"	Person4	"Blue Nile"	"London"	0547896214

يوضح الجدول (2) كيفية تخزين البيانات باستخدام تقنية جدول الخصائص متفاوت المسافات. ويتكون بشكل أساسي من عمود الفاعل وأعمدة الخصائص (N). ويحتوي عمود الفاعل على رقم الموظف وتحتوي أعمدة الخصائص على القيم أو البيانات التي تخص كل موظف.

جدول (3) فئة الخصائص (**hasAddress class**)

Subject	isNamed	hasEmployee	isCalled	hasPhone
NO1	"Ahmed Omer"	person1	"Blue Nile"	0147895478
NO3	"John Krees"	Person3	"Blue Nile"	0145258147
NO4	"James Kan"	Person4	"Blue Nile"	0547896214

يستخدم الجدول (3) تقنية جدول فئة الخصائص. ويحتوي على بيانات جميع الموظفين الذين لديهم الخاصية 'hasAddress' دون تضمين الخاصية نفسها (مثل الخاصية hasAdress تم حذفها من الجدول). الموظف الثاني (2NO) (الفاعل) "Ali Mohammed" (الكائن) لا يوجد لديه عنوان (الخاصية) وهذا هو السبب في أنه لم يظهر في الجدول.

جدول (4) فئة الخصائص (**hasPhone class**)

Subject	isNamed	hasEmployee	isCalled	has Address
NO1	"Ahmed Omer"	person1	"Blue Nile"	"KSA"
NO2	"Ali Mohammed"	person2	"Blue Nile"	"Canada"
NO4	"James Kan"	Person4	"Blue Nile"	"London"

يستخدم الجدول (4) أيضًا تقنية جدول فئة الخصائص، ولكن هذه المرة بنوع خاصية مختلف يسمى hasPhone. نفس الفكرة المطبقة في الجدول أعلاه والتي تقدم جميع بيانات جميع الموظفين مع الخاصية 'hasPhone'. الموظف رقم ثلاثة – NO3 (الفاعل) 'John Krees' (المفعول به) لا يوجد لديه رقم هاتف (الخاصية) لهذا السبب لم يظهر في الجدول.

مزايا تقنية جدول الخصائص

هناك فوائد متعددة لاستخدام تقنية جدول الخصائص: أولاً، إنه عام جدًا (يمكن تمثيل أي نوع من البيانات تقريبًا في هذا التنسيق ومن السهل تجزئة كل من قواعد البيانات العلائقية وقواعد XML في ثلاثيات RDF) ومن السهل إنشاء الأدوات التي تتلاعب ب RDF. ثانياً، يتيح إمكانية كتابة السمة. فبدلاً من تخزين المفعول به من أنواع مختلفة في عمود واحد في مخزن الثلاثيات.

عيوب تقنية جدول الخصائص

يحتوي هذا النهج على العديد من المشكلات: الأولى هي NULLs، ففي واقع الأمر، لا يمكن تعريف جميع الخصائص لجميع الفاعلين داخل تجمع الفاعل بحيث يمكن أن تحتوي الجداول الضخمة على NULLs. ويمكن أن يسيطر مقدار المسافة من هذه القيم الفارغة على مساحة البيانات نفسها. العيب الآخر هو سمات متعددة القيم: العلاقات بين أطراف كثيرة غير ملائمة إلى حد ما، للتوضيح أكثر (مثل المشروعات التي بها عدة موظفين والموظفون لديهم مشاريع متعددة). وآخر مشكلة هي خوارزمية التجميع التي يجب أن تستخدم عادة لاكتشاف تجمعات الخصائص التي تميل إلى أن تكون محددة معا أو فئات من الفاعل مع تعريفات خصائص مماثلة. علاوة على ذلك، قد تصبح إعادة التجميع ضرورية نظرًا لأن خصائص البيانات تتغير مع مرور الوقت [9].

تمت معالجة المشكلات المذكورة سابقاً، من خلال استخدام منهج التقسيم العمودي الذي ينشئ جدولًا مكونًا من عمودين لكل خاصية فريدة في مجموعة بيانات RDF حيث يحتوي العمود الأول مجموعة الفاعل القريبة التي تحدد الخاصية، بينما يحتوي العمود الثاني على قيم المفعول به لمجموعة الفاعل تلك. ومن اهم النتائج التي تم التوصل اليها إزالة القيم الخالية و حل مشكلة السمات متعددة القيم الموجودة في الجداول.

المصادر والمراجع:

[1] A Harth, M Janik, S Staab, (2011). "Semantic web architecture." Handbook of Semantic Web Technologies. Springer Berlin Heidelberg, 43-75.

[2] Buranarach, M., Supnithi, T., Thein, Y. M., Ruangrajitpakorn, T., Rattanasawad, T., Wongpatikaseree, K., ... & Assawamakin, A. (2016). OAM: an ontology application management framework for simplifying ontology-based semantic web application development. *International Journal of Software Engineering and Knowledge Engineering*, *26*(01), 115-145.

[3] Madhu, G., Govardhan, D. A., & Rajinikanth, D. T. (2011). Intelligent semantic web search engines: A brief survey. *arXiv preprint arXiv:1102.0831*.

[4] Munir, K., & Anjum, M. S. (2018). The use of ontologies for effective knowledge modelling and information retrieval. *Applied Computing and Informatics*, *14*(2), 116-126.

[5] Curé, Olivier, and Guillaume Blin, (2014). RDF Database Systems: Triples Storage and SPARQL Query Processing. Morgan Kaufmann.

[6] Álvarez-García, S., Brisaboa, N., Fernández, J. D., Martínez-Prieto, M. A., & Navarro, G. (2015). Compressed vertical partitioning for efficient RDF management. *Knowledge and Information Systems*, *44*(2), 439-474.

[7] Calbimonte, J. P., Jeung, H., Corcho, O., & Aberer, K. (2012). Enabling query technologies for the semantic sensor web. *International Journal On Semantic Web and Information Systems (IJSWIS)*, *8*(1), 43-63.

[8] Picalausa, F., Luo, Y., Fletcher, G. H., Hidders, J., & Vansummeren, S. (2012, May). A structural approach to indexing triples. In *Extended Semantic Web Conference* (pp. 406-421). Springer, Berlin, Heidelberg.

[9] Padiya, T., Ahir, M., Bhise, M., & Chaudhary, S. (2012). Data partitioning for semantic web. *International Journal of Computer & Communication Technology (IJCCT)*, 32-35.