لما لا تختلط بياناتك: عدم التجانس الدلالي

Why Your Data Won't Mix: Semantic Heterogeneity

Alon Y. Halevy

1. مقدمة Introduction

عند انشاء مخططات لقواعد بيانات لنفس النطاق او المجال "domain" بواسطة فرق برمجية مختلفة فان هذه المخططات تكون مختلفة عن بعضها البعض. ويشار إلى هذه الاختلافات بمصطلح semantic heterogeneity اي عدم التجانس الدلالي. كما تظهر عدم التجانس الدلالي ايضا في مستندات XML متعددة، وفي خدمات الويب وعلم المعلومات "الانتولوجي"، وعلى نطاق اوسع، عندما يكون هناك أكثر من طريقة لبناء مجموعة من البيانات. ان وجود بيانات شبه منظمة يفاقم عدم التجانس الدلالي. ولكي تتوافق انظمة البيانات المتعددة مع بعضها البعض فإنها يجب ان تفهم مخططات بعضها البعض. وبدون هذا الفهم، يصبح حجم مصادر البيانات للنسخ الرقمية كبيرا جدا. نبدأ في هذا المقال بمراجعة السيناريوهات المختلفة والشائعة في مجال تطبيقات مشاركة البيانات حيث يكون حل مشكلة عدم التجانس الدلالي فيها امرا ضروريا. ومن ثم نقوم بشرح لماذا من الصعب حل مشكلة عدم التجانس الدلالي، واستعراض بعض الابحاث الحديثة والتطورات على الصعيد التجاري في مواجهة هذه المشكلة. في النهاية سوف نبين المشاكل الرئيسية والفرص المتاحة للحل في هذا المجال.

2. سيناريوهات عدم التجانس الدلالي - Scenarios of Semantic Hetero- Geneity

تكامل معلومات المؤسسة: تواجه المؤسسات في يومنا هذا زيادة كبيرة في تحديات ادارة البيانات التي تتضمن بيانات الولوج (الدخول) وتحليل البيانات المتوفرة في المصادر المتعددة، مثل انظمة قواعد البيانات، وانظمة ادارة موارد المؤسسات وملفات XML والتغذية الراجعة. على سبيل المثال، لكي تتمكن مؤسسة من الحصول على عرض واحد لعميل، فإنها يجب ان تستفيد من عدة قواعد بيانات. وبالمثل ايضا عندما تقوم مؤسسة بتقديم عرض خارجي موحد لبياناتها عليها اما ان تتعاون مع جهة ثالثة او ان تقوم بإنشاء موقع ويب، وفي كلا الحالتين يجب الدخول على عدة مصادر. ومع انتشار السوق الالكترونية فان هذه التحديات تصبح عائقا امام العديد من المؤسسات.

هناك الكثير من الاسباب التي تكون فيها بيانات المؤسسات متواجدة في أكثر من مصدر ومرتبة بطرق عشوائية، ومن هذه الاسباب. أولا، الكثير من انظمة البيانات تم تطويرها بشكل مستقل لتلبية احتياجات العمل المطلوبة، لكن عندما يتطلب العمل تغيرا ما، فانه من الضروري مشاركة البيانات بين اقسام مختلفة في المؤسسة. ثانيا، تحصل المؤسسات على العديد من مصادر البيانات نتيجة للاندماج والاستحواذ بينها.

كان هناك على مر السنين العديد من الطرق لمواجهة تحديات تكامل معلومات المؤسسة. حتى اواخر التسعينات من القرن العشرين كان الحلان الاكثر انتشارا هما مستودع البيانات data warehousing وتصميم حلول متخصصة. لقد كان لحل مستودع البيانات عيب في الوصول إلى البيانات القديمة في الكثير من الحالات ولم يكن قادرا على العمل بين حدود المؤسسات. وكان الحل المتخصص مكلف جدا ومن الصعب الحفاظ عليه وغير قابل للتطوير.

قدمت في اواخر التسعينات من القرن العشرين عدة شركات حلول الاستعلام من مصادر بيانات متعددة في الزمن الفعلي. وقد استخدم المصطلح EII للإشارة إلى هذه الحلول. في حين ان مستخدمو هذه الانظمة لازالوا يشاهدون مخطط واحد (سواء كان علائقيا او XML)، حيث تترجم كل الاستعلامات على الفور إلى استعلامات ملائمة لمصادر البيانات الفردية، وتُدمج النتائج بشكل مناسب من النتائج الجزئية التي تم الحصول عليها من المصادر. والنتيجة، حصول المستخدم على اجابات معتمدة دائما على بيانات حديثة. ومن المثير للاهتمام ان العديد من الشركات قامت ببناء منتجاتها على منصات XML مستفيدة من مرونة الـ XML (وكذلك لبياناتها الشبه منظمة) والتي جعلت منها الاكثر ملائمة لتطبيقات تكامل البيانات. في مقالة حديثة [8] استعرضت الدراسة المسحية لبعض التحديات التي تواجه هذه الصناعة. وفي النهاية افترضت أحدث البحوث في هذا المجال معمارية الند للند peer-to-peer لمشاركة البيانات مع بنية ودلالات غنية [1].

في اي من معماريات مشاركة البيانات تلك، كان التوافق بين التجانس الدلالي هو المفتاح. ولا يهم إذا ما كان الاستعلام قد صدر على الفور او تم تحميل البيانات في مستودع البيانات، او سواء تمت مشاركة البيانات من خلال خدمات الويب او بطريقة الند للند، وفي النهاية فان الفروقات الدلالية بين مصادر البيانات يجب ان تتوافق مع بعضها البعض. تكون هذه الفروقات في العادة متوافقة من خلال الخرائط الدلالية *semantic mappings*. هذه عبارة عن صيغ تحدد كيف تتم ترجمة البيانات من مصدر بيانات إلى الاخر بطريقة تحافظ على دلالات البيانات، او بطريقة اخرى اعادة صياغة الاستعلام المطروح على مصدر واحد في استعلام من مصدر اخر. يمكن للخرائط الدلالية ان تُحدد في العديد من الاليات والتي تشتمل على استعلامات SQL وصيغ XQuery، وبرمجيات XSLT النصية او حتى كود Java.

من الناحية العملية تكون القضية الاساسية هي مقدار الجهد المطلوب لتحديد الخرائط الدلالية. وفي سيناريو تكامل البيانات النموذجية تستغرق عملية تحديد الخرائط حوالي نصف الجهد (وفي بعض الاحيان يصل إلى ما يقارب 80٪) وهذه العملية مرهقة وعرضة للأخطاء. معظم نواتج الـ EII في يومنا هذا تأتي ببعض الادوات لتحديد هذه التعيينات، لكن هذه الادوات يدوية ويتطلب انجازها خبراء لتحديد الخارطة المضبوطة بين مخططين.

استعلام وفهرسة الويب الخفي Deep Web: يشير مصطلح الويب الخفي إلى محتوى الويب الموجود في قواعد البيانات والتي يمكن الوصول لها من خلف النماذج. ومحتوى الويب الخفي ليس في العادة مفهرسا في محركات البحث لان زواحف crawlers هذه المحركات لا تستطيع المرور خلف النماذج. في هذا السياق يمكن رؤية النموذج كمخطط وما لم تتمكن الزواحف من فهم معنى الحقول في النموذج فإنها تعلق هناك.

مقدار وقيمة محتوى الويب الخفي مذهلة. وتتوقع بعض التقديرات ان مقدار محتوى الويب الخفي يزيد عن ١٠ او حتى ٢٠ مرة عن الويب العادي. امثلة على مثل هذا المحتوى تتنوع من الدعايات المصنفة في الاف الجرائد حول العالم إلى بيانات قواعد البيانات الحكومية وقواعد بيانات المنتجات ومستودعات الجامعات وأكثر من ذلك بكثير.

هنا ايضا ينبع التحدي من ان هناك تشكيلة كبيرة جدا لطرق تصميم مواقع الويب لأي نطاق. لهذا فانه من المستحيل على مصممو زواحف الويب افتراض شكل قياسي لتسميات الحقول والتراكيب عند قيامها بالزحف. حتى في النطاق البسيط مثل البحث عن سيارات مستخدمة فان عدم التجانس في النماذج كبيرا جدا. بالطبع يأتي التحدي الرئيسي من حجم هذه المشكلة. على سبيل المثال موقع الويب www.everyclassified.com، هو اول موقع ويب يقوم بتجميع محتواه من الاف المصادر والتي تشتمل على أكثر من 5000 خريطة دلالية لنماذج الويب ولفئات شائعة من الاعلانات المصنفة. لاحقا في المقال سوف أصف الفكرة التي جعلت من هذا الموقع ممكنا.

من المهم ان نؤكد على ان الدخول إلى الويب الخفي هو ايضا في حد ذاته تحدي لمزودي المحتوى أكثر من محركات البحث. يسعى مزودو المحتوى على جذب انتباه المستخدمين. وفي بدايات انتشار www كانت اي قاعدة بيانات جيدة تصبح معروفة على الفور على سبيل المثال IMDB للأفلام. لكن عدد قواعد البيانات المشابهة اليوم هائلا جدا (تقدر بمئات الالاف)، والناس لا تعرف عنها شيء. حيث تقوم الناس بالبحث بدءً من صندوق البحث في محرك البحث المفضل لديهم، وهذه المحركات تقوم بعمل سيء جدا لفهرسة محتوى الويب الخفي. وعليه، إذا قمت بصناعة قاعدة بيانات ممتازة لوصفات الشرق الاوسط ووضعتها على شبكة الويب خلف النماذج فإنها ستبقى خفية. ومن المفارقات ايضا انني أفضل انشاء مجموعة من صفحات الويب التي تحتوي على محتويات وصفة من انشاء قاعدة بيانات قابلة للبحث. في النهاية يجب ان نذكر ان البحث في قواعد بيانات المؤسسات يواجه مشكلة مشابهة بعض الشيء: فالكثير من مصادر قواعد البيانات الهامة داخل المؤسسة تكون في قواعد بيانات وتوفير كلمات مفتاحية بسيطة لهذا المحتوى هو تحدي كبير.

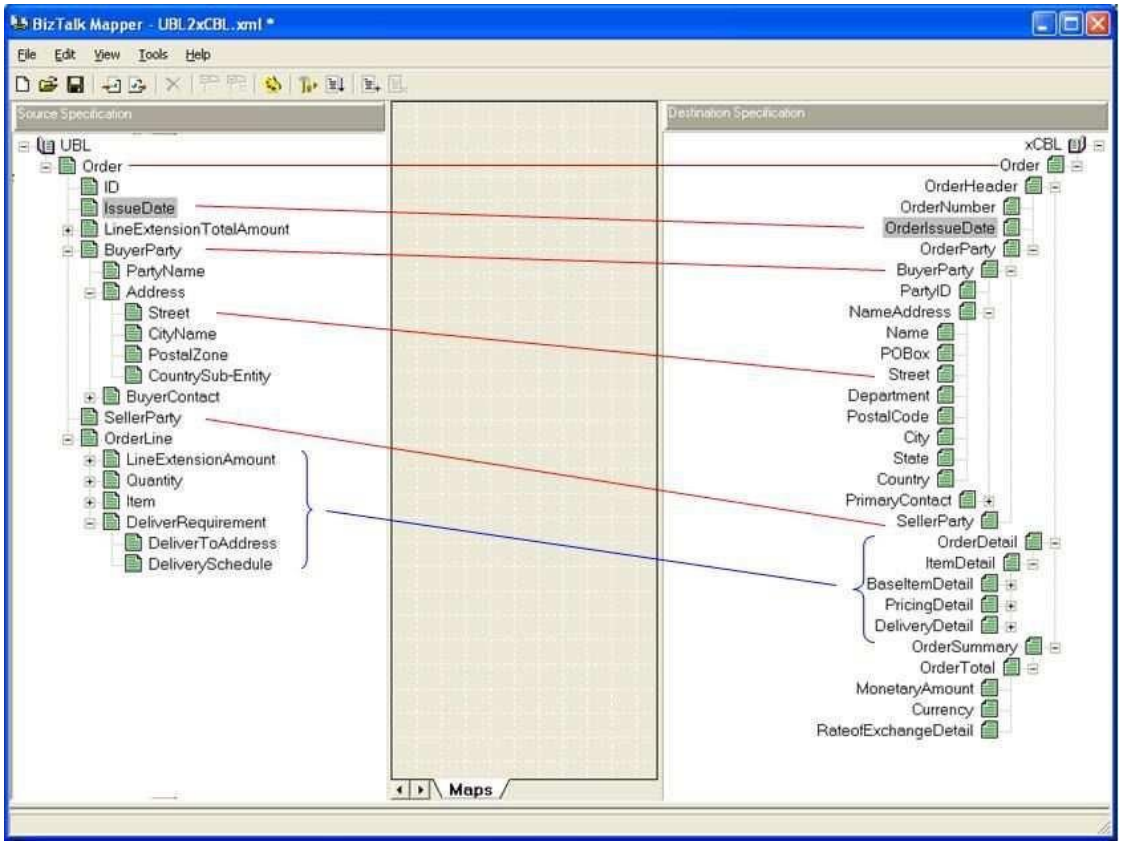
تعيين كتالوج تجاري Merchant Catalog Mapping: مثال مألوف لعدم التجانس الدلالي يحدث في تجميع كتالوج المنتج.

اعتبر تاجر تجزئة الكتروني مثل موقع امازون amazon.com. مثل هذا التاجر يقبل تغذية المنتجات من الالف التجار، وكل تاجر منهم يحاول بيع بضاعته الكترونيا. لتجميع كل هذا العدد الهائل من المعلومات يقوم كل تاجر بوصف مخططه الذي يشتمل على التسلسل الهرمي للمنتجات والسمات المصاحبة له. لكن على الجانب

الاخر تخزن البيانات عند التاجر على مخططه المحلي، والذي يكون مختلفا عن مخطط تاجر التجزئة (وعادة ما يغطي جزء صغير من هذا المخطط). وعليه فان المشكلة التي نواجهها هنا هو انشاء الخرائط بين الاف التجار والعدد المتزايد من التجار على الانترنت (ما يقارب 10 في الولايات المتحدة في نفس الوقت). والنقطة الهامة حول هذا السيناريو هو انه ليس بالضرورة وجود مخطط دلالي واحد من التاجر إلى تاجر التجزئة. وعوضا عن ذلك، وحيث ان هناك فروقات دقيقة بين فئات المنتج وان المنتجات يمكن تدرج في اكثر من فئة فان هناك خرائط متعددة قد تكون منطقية وافضلها هي تلك التي تبيع منتجات أكثر!

المخطط مقابل عدم تجانس البيانات Schema versus Data heterogeneity: لا يحدث عدم التجانس فقط في المخططات لكن ايضا في قيم البيانات الفعلية نفسها. على سبيل المثال، هناك طرق متعددة للإشارة إلى نفس المنتج. وعليه حتى إذا اخبرك أحد ان حقل معين في خرائط بيانات التاجر هو ProductName والذي قد لا يكون كافيا لحل مشكلة المراجع المتعددة لمنتج واحد. ومن الامثلة الشهيرة الاخرى، هناك عدة طرق للإشارة إلى الشركات (على سبيل المثال IBM مقابل International Business Machines) واسماء الناس (والتي في الاغلب تكون غير كاملة) وكذلك العناوين. يتطلب الدمج الكامل للبيانات من مصادر متعددة التعامل مع كلا من مستوى عدم التجانس الدلالي ومستوى عدم تجانس البيانات. عادة ما تواجه المنتجات المختلفة هذين الجزئيين من المشكلة بشكل منفرد. على سبيل المثال، ركزت العديد من المنتجات لتحليل الانفاق العالمي على مستوى عدم التجانس الدلالي. وتركز هذه المقالة بالأخص على عدم تجانس المخطط

عدم تجانس المخطط والبيانات شبه المنظمة Schema heterogeneity and semi-structured data: انني اقول ان مشكلة عدم التجانس الدلالية تتفاقم عندما نتعامل مع بيانات شبه منظمة، وذلك لعدة اسباب. اولا، ان التطبيقات التي تتضمن بيانات شبه منظمة تكون في العادة هي المستخدمة في مشاركة البيانات بين عدة أطراف، وعليه يكون عدم التجانس الدلالي جزء من المشكلة منذ البداية. ثانيا، تكون المخططات شبه المنظمة أكثر مرونة وبذلك يكون هناك احتمالية أكثر لرؤية اختلافات في المخطط. واخيرا، الميزة الاساسية للبيانات شبه المنظمة هو امكانية اضافة المزيد من السمات إلى البيانات (او حتى ببساطة استخلاصها من فحص البيانات)، وبمجرد ان تكون المرونة متوفرة فان عدد السمات المضافة تكون مهمة، ويصبح فهم معناها بدقة امرا حساسا. على الجانب الاخر، في الكثير من التطبيقات التي تشتمل على بيانات شبه منظمة فانه من الكافي عمل توافق مع مجموعة محددة من الخصائص فقط، بينما يمكننا ان نغير ونستعرض اي خاصية اخرى. وعلى وجه التحديد، فإننا نحتاج فقط لتوفيق هذه الخصائص والتي ستستخدم لمساواة البيانات عبر المصادر المتعددة.



الشكل 1 التوافق بين مخططين متباينين (مختلفين).

3. لماذا الامر صعبا إلى هذا الحد؟ Why is it so hard?

لقد كانت مشكلة توافق المخطط الدلالي موضوع البحث لعدة عقود، لكن الحلول المتوفرة قليلة. السبب الرئيسي الذي يجعل من عدم التجانس الدلالي صعبا إلى درجة كبيرة يعود إلى ان مجموعات البيانات قد تم تطويرها بشكل مستقل، ولهذا استخدمت الكثير من التراكيب لتمثيل نفس المفاهيم وربما تداخلت مع بعضها البعض. ونحاول في الكثير من الحالات، دمج انظمة البيانات التي تم تطويرها لتلبية احتياجات الاعمال المختلفة والتي قد يكون الاختلاف بينها صغيرا او حتى كبيرا جدا. وبالتالي، حتى إذا تمت نمذجة نطاقات متداخلة فان هذه النمذجة سوف تكون بطرق مختلفة. التراكيب المختلفة هي نواتج طبيعة الانسان – من المعروف ان الاشخاص يفكرون بطرق مختلفة عن بعضهم البعض حتى عندما يواجهون نفس الهدف. للتوضيح سوف نضرب مثالا على ذلك، في واحدة من الواجبات المنزلية التي اعطيتها في مقرر قواعد البيانات كانت حول تصميم مخطط مصنع بالاعتماد على وصف انجليزي لما يجب ان يحتويه. بدون اي تغير كانت المخططات التي حصلت عليها من الطلبة مختلفة بشكل كبير [6].

من وجهة نظر عملية، فان واحدة من الاسباب التي يعود لها صعوبة عدم تجانس المخطط والاستنزاف الزمني

(تتطلب وقت كبير) هو انها تحتاج لكلا من النطاق والخبرة التقنية: تحتاج إلى شخص يفهم معنى ادارة الاعمال لكل مخطط ومهارات الناس في كتابة التحويلات (على سبيل المثال خبراء في SQL او XQuery).

مع ان موضوع عدم تجانس المخطط يعد تحديا للإنسان، فهو يعتبر أكثر تحديا للبرامج. يعطى للبرنامج مخططين فقط ليقوم بإجراء التوافق بينهما - لكن هذه المخططات هي في الحقيقة مجرد رموز. انها لا تلتقط كامل المعنى او المقصود في المخططات – تلك الامور تكون في عقول المصممين فقط.

يشرح الشكل 1 بعض التحديات التي تواجه حل عدم التجانس الدلالي. يوضح الشكل اداة مطابقة يدوية بحيث يحتاجها المصمم لرسم خطوط بين الخصائص المتطابقة بين المخططين. كما هو موضح في المثال، هناك عدة انواع من الاختلافات الدلالية بين المخططين (1) نفس عناصر في المخططين اعطيت اسماء مختلفة (على سبيل المثال IsueDate وOrderIssueDate)، (2) الخصائص في المخطط مجمعة في جداول او في XML بطرق مختلفة (على سبيل المثال اعتبر شجرة فرعية لعنصر المشتري BuyerParty element في المخططين) و(3) تغطي أحد المخططين جوانب النطاق في حين انها لم تغطى بالمخطط الاخر (على سبيل المثال، المخطط على اليسار لا يمتلك اي شيء مثل OrderSummary في المخطط على اليمين.

عند توفيق عدم التجانس من الاف نماذج مواقع الويب يكون هناك مصادر اضافية لعدم التجانس. بعض النماذج بالفعل تكون متخصصة لنطاق محدد (موقع سيارات مستخدمة او موقع وظائف)، في حين ان الاخرى يتطلب على المستخدم اختيار فئة قبل ادخال خصائص جديدة. في بعض الحالات يكون الموقع ضمني في النموذج (مثل استخدام حقول خفية)، في حين ان الاخرى على المستخدم اختيار المدينة او الدولة او المنطقة البريدية.

في الاغلب يكون الطريق لحل عدم التجانس الدلالي من خلال المخططات القياسية. على اي حال، بينت الخبرة العلمية ان المخططات القياسية لها نجاحات محدودة جدا وفقط في النطاقات التي يتم الاتفاق على حوافز مجدية لاستخدام المخططات القياسية. وحتى مع ذلك فانه مع امكانية قيام مزود البيانات بمشاركة بياناتهم باستخدام النماذج القياسية فان انظمة بياناتهم لا تزال تستخدم نماذجها الاصلية (وتكلفة تغير انظمتهم باهظة جدا). وعليه يتوجب ايجاد حل لعدم التجانس الدلالي في الخطوة التي يظهر فيها مزود البيانات بياناته لنظرائه.

كما يفكر اي شخص في تقديم حل لمشكلة عدم التجانس الدلالي فانه من المهم ملاحظة العديد من الحالات الشائعة للمشكلة، والتي من الممكن ان تلقي الضوء على المشكلة الاساسية الجاري حلها:

- مخطط ما قد يكون عبارة عن نسخة جديدة عن مخطط سابق.
- يمكن انشاء مخططين من نفس المخطط الاصلي.
- قد يكون لدينا الكثير من المصادر التي تنمذج نفس الجوانب للنطاق الاساسي (التكامل الافقي).
- قد يكون لدينا مجموعة من المصادر التي تغطي نطاقات مختلفة لكنها تتداخل عند الوصلات (تكامل رأسي).

4. حل ابتكاري The state of the art

حل مخطط عدم التجانس بطبعه عبارة عن عملية تتم بمساعدة الانسان. ما لم يكن هناك ضوابط قوية على كيفية اجراء التوافق بين مخططين مختلفين عن بعضهما البعض، فلا أحد يأمل ان يحصل على حل أتوماتيكي كامل. الهدف هو تقليل الزمن المستغرق بواسطة الشخص الخبير لعمل خارطة بين زوج من المخططات، وتمكنهم من التركيز على الاجزاء الصعبة والغامضة من الخارطة. على سبيل المثال، الادوات التي تمكن من بناء موقع الويب www.everyclassified.com تطلبت منا ان نكون قادرين على رسم خارطة حقول موقع الويب من مخططنا في دقيقة واحد في المتوسط.

كما هو متوقع، حاول الناس بناء انظمة مطابقة المخططات شبه أتوماتيكية من خلال استخدام مجموعة متنوعة من الاستدلالات (انظر [13] لإجراء المسح). فيما يلي نقدم مراجعة لبعض من هذه الانظمة وقصورها. لاحظنا ان عملية اجراء التوفيق لعدم التجانس الدلالي يتضمن خطوتين. نطلق على الخطوة الاولى تطابق المخطط حيث نقوم بإيجاد التطابقات بين زوج (او مجموعات أكبر) من عناصر مخططين يشيران إلى نفس المبادئ او الاشياء في العالم الحقيقي. في الخطوة الثانية نقوم ببناء هذه التطابقات لتشكيل صيغ المخطط الفعلي. ويعد مشروع سيلو Cilo Project في شركة IBM [15] مثال رئيسي على العمل على بناء صيغ المخطط.

الفئات التالية لاستدلال والتي استخدمت في عملية تطابق المخططات.

اسماء عناصر المخطط Schema element names: اسماء العناصر (على سبيل المثال الجدول واسماء السمات او الخصائص) التي تحمل بعض المعلومات حول دلالاتها. وبالتالي من خلال البحث عن الاسماء والتي تكون في اول الكلمات يمكننا ان نحصل على ادلة لمطابقة المخطط. ان التحديات التي تتعلق باستخدام الاسماء كونها تستخدم مرادفات شائعة مثل استخدام الاسماء الهجينة (كلمات متخصصة او عامة). علاوة على اننا غالبا ما نرى نفس الكلمة تستخدم بمعاني مختلفة. كما اننا في الغالب نرى اختصارات وارتباطات تظهر في اسماء العناصر.

انواع البيانات Data types: عناصر المخطط المرتبط الذي يحدد خريطة تجمع المخططات مع بعضها البعض يحتوي في الاغلب على انواع بيانات متوافقة ولكن هذا بالطبع ليس القاعدة. على اي حال في الكثير من المخططات تكون انواع البيانات غير محددة مثل CDATA لـ XML. في الواقع العملي تعتبر انواع البيانات ذات دلالة مفيدة لاستبعاد مجموعة محددة من التطابقات

حالات البيانات Data instances: عناصر من مخططين متطابقين مع بعضهما البعض في الغالب لهما قيم بيانات متشابهة. تنشأ التشابهات بطرق مختلفة: (1) القيم المستخلصة من نفس النطاق مثل صانع السيارات او اسماء الدول، (2) احداث مهمة لها نفس القيم مثل التفضيلات التي تصف منازل للبيع او (3) نمط القيم مثل

ارقام الهواتف او مدى الاسعار. حالات البيانات مفيدة جدا عندما تكون متاحة ولكن لا يمكن الاعتماد على توفرها.

بنية المخطط Schema structure: عناصر المطابقة في المخطط تكون في الاغلب مرتبطة مع عناصر مخططات اخرى ذات علاقة. على سبيل المثال في هرمية الكائن الموجه object-oriented، فانه في الاغلب إذا فئتين تتطابقان مع بعضهما البعض فان اتباع (اطفال) هاتين الفئتين سوف تكونان ايضا (او جزئيا) متطابقتين. في XML يكون تقارب DTDs (اختصار لـ تعريف نوع المستند) للسمات في DTD مثل حقل الهاتف بجوار الوكيل يقترح ان هذا الهاتف لهذا الوكيل. على اي حال الاعتماد على مثل هذه الدلالة يمكن ان يكون هشا، ومن التحديات الرئيسية هو ايجاد التطابق الاولي الذي يستخلص التشابهات بين الحقول المتجاورة.

قيود التكامل Integrity constraints: اعتبار قيود التكامل على سمة واحدة او عبر كل السمات من الممكن ان يكون مفيدا لتوليد التطابقات. على سبيل المثال إذا كانت سمتين معروفتين على انهما مفاتيح في مخططاتهما فانهما يوفران دليلا اضافيا للتشابه.

في حين ان كل من هذه الدلالات مفيدة فقد تبين مع الخبرة ان الاعتماد على اي منهم بصورة منعزلة يؤدي إلى مخطط الوصول لتطابق هش. وعليه فقد ركز البحث العلمي على بناء انظمة تعمل على دمج أكثر من دلالة معا [2,3.12].

بالرغم من هذه الافكار والمنتجات التجارية تعتمد بشكل كامل على كتيب توصيف الخرائط الدلالية. فإنها تساعد على توفير تداخلات مرئية تمكن المصممين من رسم الخطوط بين عناصر المخططات المتباينة، في حين ان التفاصيل الجوهرية للخرائط تتولد في النهاية في اغلب الاحيان. عملت هذه الادوات على توفير الوقت بشكل كبير ولكنها لا تقترح المخططات للمصممين.

5. حل مستجد: الاستفادة من الخبرة السابقة An emerging solution: Leveraging past experience

من أحد الاسباب الرئيسية التي يعود لها هشاشة الحلول المذكورة سابقا حول تطابق الخطة هو انها تستغل الادلة الموجودة في المخططين المستخدمين في عملية التطابق بينهما واهمال الخبرة السابقة. تعاني هذه المخططات في الاغلب من نقص في الادلة التي تمكننا من اكتشاف التطابقات. على اي حال، بالتدقيق أكثر في مهام تطابق المخططات فانه من المؤكد ان هذه المهام هي في الاغلب تكرارية. بشكل محدد أكثر، اننا نجد نقوم بتكرار رسم المخططات في نفس النطاق من خلال مخطط وسيط. على سبيل المثال عمل المحرك في موقع www.everyclassified.com يتضمن رسم الاف نماذج الويب في نفس النطاق من خلال مخطط عام، وهو

المتاح للمستخدمين. الخبرة البشرية بعد رؤية العديد من المخططات في نطاق معين تمكنهم من رسم المخططات بشكل أسرع لأنهم يرون الكثير من التغيرات حول كيف تم تمثيل اساسيات ومبادئ النطاق في المخططات.

التحدي هنا هو اعادة عمل مطابق المخطط بنفس الامكانيات والقدرات: بالاستفادة من الخبرة السابقة. على سبيل المثال بمجرد ان اعطي للنظام عدة خرائط في نطاق السيارات المستخدمة فانه يجب ان يكون قادرا على التنبؤ بخرائط المخططات التي لم يتم مشاهدتها من قبل. كما ترى مخططات أكثر في نطاق محدد فان توقعاته يجب ان تكون أكثر دقة وأكثر متانة في وجود التغيرات.

تم استكشاف هذه الفكرة في السنوات القليلة الماضية بواسطة مجموعة من الأكاديميين [3,7,9,10,11]، وحديثا تم تطبيقها تجاريا لأول مرة بواسطة Transformic Inc. مؤسس موقع www.everyclassified.com.

المشاريع البحثية التي تستخدم تعلم الالة Machine Learning كألية لتمكين مطابق المخطط من الاستفادة من الخبرة السابقة. يوفر النظام في تقنية تعلم الالة مجموعة من الامثلة التدريبية، وتستخدمها لتعلم نماذج النطاق تحت الدراسة. في هذا السياق تكون خرائط المخطط التي انشئت يدويا بواسطة خبراء النطاق واعطيت للنظام. تمكن نماذج النطاق النظام من النظر إلى مخططات جديدة وتتوقع من خلالها خرائط المخطط. على سبيل المثال يمكن للنظام ان يتعلم السمة الخاصة بمواصفات المنزل والتي تشتمل على نص طويل واحداث متكررة من التفضيلات. علاوة على ان النظام يمكنه ان يتعلم الاختلافات في طريقة تسمية الناس لهذا الحقل.

محرك بحث خدمة الويب Web-service search engine: تطبيق اخر لهذه الفكرة هو البحث عن خدمات الويب، اي تحديد موقع خدمات الويب (او العاملين في هذا المجال) والتي لها علاقة بحاجة خاصة. لا يعتبر كلمة مفتاح البحث البسيطة كافية في هذا السياق لان الكلمات المفتاحية (او عوامل التسمية) لا تقتنص الدلالة من وراء خدمة الويب. محرك البحث Woogle (موضح في المقالة [4] ومتاحة على الموقع (www.cs.washington.edu/woogle) تعتمد على تحليل مجموعة كبيرة من خدمات الويب معاملات الاسماء في مفاهيم دلالية لها معنى. تستخدم هذه المفاهيم للتوقع عندما يكون هناك مشغلين خدمة ويب لهما نفس الوظيفة.

6. ماذا يمكنك ان تتعلم من الماضي؟ What can you learn from the past?

نموذج التعلم من الخبرة السابقة لإنجاز مهمة مطابقة الخطة لازالت في مراحلها الابتدائية. ومن المهم ان نأخذ خطوة للخلف ونعتبر ما يمكن ان يتعلمه أحد من الماضي في هذا السياق.

لقد افترضنا ان الماضي يعطى لنا كمجموعة من الخطط لنطاق محدد إلى اقصى حد ممكن، مع الخرائط بين ازواج الخطط في تلك المجموعة، وايضا حالات البيانات. يمكن ان تأتي المخططات من اي مكان ويمكن ان تحتوي على نطاقات لها صلة مع بعضها البعض، وليس بالضرورة تنمذج نفس البيانات. في الكثير من الحالات مثل المخططات يمكن الحصول عليها من الويب او مصادر مثل xml.org. وفي اخرى، قد تكون متوفرة من

خلال المؤسسة. ونشير في الغالب لمثل هذه التجمعات من المخططات على انها جسم "corpus" بالقياس باستخدام جسم مستندات استرجاع المعلومات الاساسية ومحركات بحث الويب. بالطبع بينما الجسم في مستندات استرجاع المعلومات الاساسية يحتوي على مجموعات من الكلمات، هنا يكون لدينا عناصر دلالية غنية مثل المخططات وحالاتها.

الهدف من تحليل جسم المخططات وخرائطها هو توفير تلميحات مساعدة عن مفاهيم النطاقات الخفية وبأكثر دقة ممكنة. بالنظر عن قرب على هذه الطريقة فان الامثلة التالية لما يمكن ان نتعلمه من الجسم هو على النحو التالي:

مفاهيم النطاق واختلافاتها التمثيلية (الدلالية): كخطوة اولية يمكننا ان نحلل الجسد ونعرف المفاهيم الرئيسية في النطاق. على سبيل المثال، في جسم مخطط مستودع كتب، يمكننا ان نعرف مبدأ الكتاب والمخزن ومجموعات الاسعار المرتبطة بالعناصر. والاكثر اهمية من ذلك اننا سوف نكتشف التغيرات المتعلقة بكيف تم تمثيل هذه المفاهيم. يمكن للاختلافات ان تختلف بتسمية عناصر المخطط، وسمات التجميع في جداول او دقة نمذجة مبدأ محدد. معرفة هذه المتغيرات سوف يكون ميزة عندما نقوم بإجراء تطابق بين مخططين في النطاق.

العلاقات بين المفاهيم: من خلال مجموعة من المفاهيم المعطاة يمننا ان نكتشف العلاقات بينهم، وكذلك الطرق التي من خلالها ظهرت هذه العلاقات. على سبيل المثال يمكننا ايجاد ان جدول الكتب يحتوي على عمود ISBN ومفتاح خارجي في الجدول المتوفر، لكن لا يظهر ISBN في جدول المستودع. هذه العلاقات مفيدة في تقليص المخططات المتطابقة المرشحة التي تظهر بأقل احتمالية. هذه يمكن ان تستخدم ايضا لبناء نظام يوفر النصيحة في تصميم مخططات جديدة.

قيود النطاق: يمكننا الاستفادة من الجسم corpus لإيجاد قيود التكامل على النطاق وما يمثله. على سبيل المثال يمكننا ان نلاحظ ان ISBN هو مفتاح خارجي في عدة جداول تشتمل على كتب، وبالتالي من الممكن ان نتكون معرفا للكتب، او اكتشاف انواع البيانات المحتملة لحقول معينة (مثل العنوان والسعر). من الممكن ان تقوم القيود بترتيب السمات. على سبيل المثال في جسم نماذج موقع ويب عن سيارات للبيع، من الممكن ان نكتشف ان سمة الصانع او المنتج دائما ما تكون قبل سمة الموديل وسمة السعر، لكن بعد سمة جديد او مستخدم.

في العادة، القيود التي اكتشفناها في هذه الطريقة هي قيود برمجية بمعنى انها في بعض الاحيان تنتهك، لكنها لا تزال تعتبر من البديهيات حول النطاق. لهذا هناك فائدة عظمى من حل الحالات الغامضة، مثل الاختيار من بين عدة الخطط المرشحة المتطابقة.