

- ническая информатика и Телемедицина» 2004. – №2. – С.137–146.
2. Кальниш В.В. «Роль інформаційних технологій у інтелектуалізації охорони здоров'я»//«Клиническая информатика и Телемедицина» 2004. – №1. – С.28–34.
 3. Нильссон Дж. Применение DDD и шаблонов проектирования: проблемно-ориентированное проектирование приложений с примерами на С# и .NET: Пер. с англ. М.: ООО «И.Д. Вильямс», 2008. – 560 с.
 4. Papa J. The Entity Framework In Layered Architectures. //MSDN Magazine, 2008. <http://msdn.microsoft.com/en-us/magazine/cc700340.aspx>.
 5. Андерсон К. Основы Windows Presentation Foundation. Пер. с англ. А. Слинкина – М.: ДМК Пресс, 2008. – 432 с.
 6. Кренке Д. Теория и практика построения баз данных. СПб.: Питер, 2003. – 800 с.

УДК 004.82

ОПРАЦЮВАННЯ КОНТЕКСТУ У КОГНІТИВНІЙ ІНФОРМАЦІЙНІЙ СИСТЕМІ КЕРОВАНІЙ МОДЕЛЯМИ

Є.В. Буров

Кандидат технічних наук, доцент
Кафедра „Інформаційні системи та мережі”
Національний університет „Львівська політехніка”
вул. Ст.Бандери, 13, м. Львів
Контактний тел.: (032) 258-2538
E-mail: eugene_burov@yahoo.com

Опрацювання контекстних даних є важливою вимогою до когнітивних інформаційних систем. У статті запропоновано формальне визначення поняття контексту факту та класу бази знань. Розглянуто питання пошуку фактів, навчання та видобування знань з контексту

Ключові слова: контекст, модель, когнітивна система

Использование контекстных данных является важным требованием к когнитивным информационным системам. В статье предложено формальное определение понятия контекста факта и класса базы знаний, рассмотрены проблемы поиска информации, обучения и формирования новых знаний в пределах определенного контекста

Ключевые слова: контекст, модель, когнитивная система

Processing and using context information is an important requirement for cognitive information system. Paper proposes a formal context definition and discusses information search, learning and knowledge mining within context

Keywords: context, model, cognitive system

Постановка проблеми

Поняття контексту є одним з центральних концептів, дослідження яких необхідно для побудови когнітивної інформаційної системи керованої моделями [1]. Моделі під час взаємодії обмінюються відносно невеликими обсягами інформації, передаючи тільки посилення, 'опорні точки' у семантичній мережі взаємопов'язаних фактів. Усю недостаючу для прийняття рішення інформацію модель повинна отримати самостійно, з контексту. Таким чином, зміст та обсяг

потрібної для прийняття рішення інформації визначається активованою моделлю, а не моделлю активатором, що уможлиблює визначення різних моделей для досягнення однієї і тої ж мети. Вибір та використання конкретної моделі залежить від контексту.

Не неформальному, інтуїтивному рівні взаємодія двох об'єктів, яка відбувається з використанням певного протоколу, значно спрощується, якщо обидва об'єкти мають доступ до спільного джерела інформації, що містить релевантну до процесу взаємодії інформацію. До цього джерела учасники взаємодії

звертаються, якщо інформація отримана шляхом обміну є недостатньою для прийняття рішення. Частина інформаційного масиву джерела, яка є релевантною до задачі, що вирішується в процесі обміну даними і є контекстом цієї задачі.

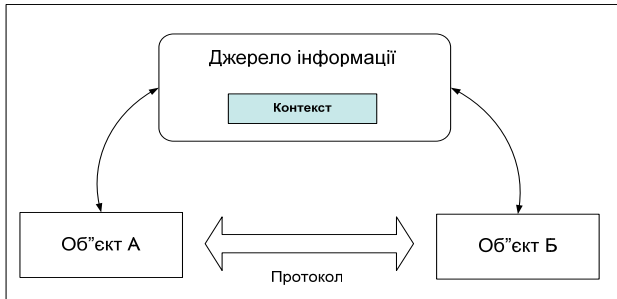


Рис. 1. Використання контексту у процесі взаємодії об'єктів

Енциклопедичні та словникові джерела дають таке визначення терміну 'контекст'. Контекст – це „взаємопов'язані умови в яких щось існує або відбувається” (interrelated conditions in which something exists or occurs) [2].

Поняття 'контекст' широко застосовується у лінгвістиці, комп'ютерних науках, зокрема в таких нових наукових напрямках як повсюдний комп'ютинг (Ubiquitous computing), автономні обчислення, контекстно-залежні сервіси.

Різні автори по-різному розуміють та дають різні визначення цього терміну, залежно від напрямку та мети проведених ними досліджень. Так, [3] визначає контекст як 'довільну інформацію, яка може бути використана для характеристики ситуації певної сутності. Такою сутністю може бути об'єкт, місце, чи особа яка є релевантною для взаємодії користувача та застосування, включаючи самого користувача та застосування'.

В [4] контекст визначено як знання про стан користувачів та пристроїв, в тому числі про оточуюче середовище. В [3] конкретизовано такі аспекти контексту як розміщення об'єкту, його ідентифікацію та дії, час.

В [5] відмічено такі особливості та аспекти контексту як;

- типи контексту:
- контекст користувача-людини, включаючи такі характеристики як ідентифікацію, розміщення, мобільність, доступні пристрої, та ін.
- контекст пристрою, у тому числі ір -адресу, інші конфігураційні параметри
- контекст мережі (ідентифікатор, ресурси, порти, рівень безпеки)
- контекст потоку (вузли мережі, пов'язані потоком, затримки, надійність та безпека)
- постійність контексту- виділення постійної та змінної частини контексту
- релевантність для визначеного сервісу - необхідний та опціональний

- часові аспекти: контекстна інформація з минулого, теперішнього часу та майбутнього.

В [5] визначено контекстно- залежний комп'ютинг як парадигму комп'ютингу в якій застосування та сервіси використовують таку інформацію з контексту як розміщення користувача та пристроїв, час доби, інформацію про близьких географічно користувачів та пристроїв. Водночас, в [5] неоднаразово підкреслено що існуючі визначення контексту або занадто загальні та інтуїтивні (і таким чином не можуть бути з користю використані для вирішення практичних задач) або занадто специфічні (і таким чином придатні тільки для вирішення окремих конкретних задач).

В існуючих підходах до визначення контексту для контекстно - залежних сервісів зміст контексту визначається вручну, проектувальником та таким чином слабо пов'язаний з задачею яка вирішується у системі, і не відображає всіх релевантних об'єктів. Контекстна інформація про стан середовища створюється зовнішніми сервісами (context widgets), зчитується по запиті і використовується застосуванням при потребі.

Враховуючи динамічний характер оточення, з якого отримують інформацію про контекст, ступінь релевантності інформації постійно змінюється, що також не може бути враховано у підходах, які зміст контексту визначають заздалегідь, під час проектування сервісу.

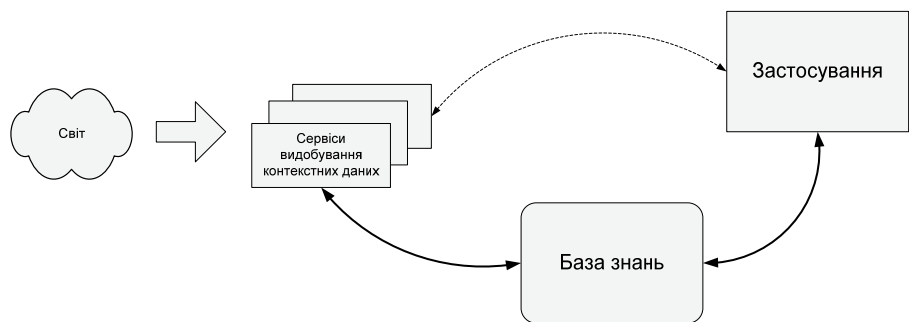


Рис. 2. Використання бази знань у застосуваннях, що працюють з контекстом

На нашу думку зазначені проблеми можна вирішити шляхом дотримання таких архітектурних принципів та рішень.

1. Використати базу знань як проміжний рівень між сервісами, що добувають інформацію з зовнішнього світу та застосуванням (рис. 2). Замість використання контексту певного реального об'єкту використовувати контекст відповідного об'єкту бази знань.

2. Запровадити семантичну інтерпретацію контексту як частини цієї бази знань, яка відображає стан предметної області. Це дасть змогу визначити контекст автоматично залежно від конкретних об'єктів, мети та потреб конкретних задач та бieżучого стану бази знань, а також розширити поняття контексту включив у нього потенційно усі типи об'єктів предметної області.

3. Відокремити контекст від задачі, яка вирішується з застосуванням контексту. Таким чином контекст певного об'єкту визначається тільки цим об'єктом, його властивостями і зв'язками з зовнішнім середовищем.

4. Визначити, що застосування, яке використовує контекст певного об'єкту, формулює критерії релевантності інформації з контексту, стратегії пошуку

релевантної інформації у контексті та спосіб використання отриманих результатів пошуку.

Формальна специфікація контексту

Розглянемо формальне визначення контексту як частини бази знань.

Нехай розглядається контекст об'єкту (факту) F_{co}

Тоді контекст цього об'єкту $Con(F_{co})$ є частиною бази знань:

$$Con(F_{co}) = BdKn_{co} \subseteq BdKn = \{M(Fc), M(LnFc)\}$$

Зв'язок між фактами задано на структурі фактів $SqLnFc$, які він поєднує:

$$LnFc = \{SqLnFc, M(PrLnFc)\}$$

де $M(PrLnFc)$ – множина значень властивостей зв'язку.

Тоді контекстом рівня 0 буде сам факт та порожня множина його зв'язків:

$$Con^0(F_{co}) = \{M_{co}^0(Fc), M_{co}^0(LnFc)\} = \{F_{co}, \emptyset\}$$

Контекстом першого рівня буде об'єднання контексту рівня 0 та множини фактів, безпосередньо пов'язаних з F_{co} та зв'язків між ними:

$$Con^1(F_{co}) = \{M_{co}^1(Fc), M_{co}^1(LnFc)\}$$

$$\text{де } M_{co}^1(Fc) = M_{co}^0(Fc) \cup M_{co}^1(Fc)'$$

$$\forall Fc_i \in M_{co}^1(Fc) \exists LnFc_j : Fc_{co} \in SqLnFc_j, Fc_i \in SqLnFc_j$$

$$\forall LnFc_i \in M_{co}^1(LnFc) : \forall Fc_j \in SqLnFc_i : Fc_j \in M_{co}^1(Fc)$$

Контекст рівня k визначимо рекурсивно

$$Con^k(F_{co}) = \{M_{co}^k(Fc), M_{co}^k(LnFc)\}$$

$$\text{де } M_{co}^k(Fc) = M_{co}^{k-1}(Fc) \cup M_{co}^k(Fc)'$$

$$\forall Fc_i \in M_{co}^k(Fc) \exists (LnFc_j, Fc_l \in M_{co}^{k-1}) : Fc_i \in SqLnFc_j, Fc_l \in SqLnFc_j$$

$$\forall LnFc_i \in M_{co}^k(LnFc) : \forall Fc_j \in SqLnFc_i : Fc_j \in M_{co}^k(Fc)$$

Розмір бази знань є скінченним та обмежує кількість рівнів контексту, які можна побудувати в ній. Нехай контекст рівня m – це максимальний рівень контексту який можна побудувати для даної бази знань та об'єкту F_{co} .

Визначимо поняття контекст як контекст максимального рівня m:

$$Con(F_{co}) = Con^m(F_{co}) \perp m = \max(k) \tag{1}$$

Якщо розмір бази знань є великим, визначення контексту вимагатиме значних витрат часу та ресурсів. З іншого боку, чим менше рівень контексту, тим семантично 'ближчими' є факти та зв'язки, що входять у

нього. Тому, для розв'язку практичних задач доцільно обмежитися декількома рівнями контексту.

З використанням визначення контексту (1) сформулюємо контекст певної моделі Md_{co} як об'єднання контекстів усіх об'єктів, що входять в неї:

$$Con(Md_{co}) = \bigcup Con(F_{co}) \perp F_{co} \in Md_{co}$$

З практичної точки зору, для виявлення досяжності мети моделі Gf важливим і корисним є поняття контексту функції мети, яке ми визначимо як сукупність фактів та вимог до них, що які використовуються функцією мети $GF(StBdFc)$:

$$Con(GF(StBdFc)) = \{M(Fc_i), M(Asr_j(StBdFc))\} \perp \forall i : Fc_i \in GF \wedge \forall j : Asr_j \in GF$$

Визначення контексту, відображене у формулі (1) наведено стосовно певного факту F_{co} . Контекст факту відображає наявну у базі знань інформацію стосовно наявних пов'язань факту контекст якого розглядається. Контекст на рівні фактів використовується для прийняття рішення та розуміння з використанням моделей, планування взаємодії з контекстними сервісами. Для вирішення задач навчання та видобування знань у певному контексті корисним є поняття контексту визначене на рівні знань.

Центральним елементом контексту на рівні знань є певний клас Cl_{co} .

Контекстом знань нульового рівня буде сам клас та порожня множина його зв'язків:

$$Con_{kn}^0(Cl_{co}) = \{M_{co}^0(Cl), M_{co}^0(LnCl)\} = \{Cl_{co}, \emptyset\}$$

Контекст знань першого рівня містить усі зв'язки та класи з якими клас контексту поєднаний безпосередньо:

$$Con_{kn}^1(Cl_{co}) = \{M_{co}^1(Cl), M_{co}^1(LnCl)\}$$

$$\text{де } M_{co}^1(Cl) = M_{co}^0(Cl) \cup M_{co}^1(Cl)'$$

$$\forall Cl_i \in M_{co}^1(Cl) \exists LnCl_j : Cl_{co} \in SqLnCl_j, Cl_i \in SqLnCl_j$$

$$\forall LnCl_i \in M_{co}^1(LnCl) : \forall Cl_j \in SqLnCl_i : Cl_j \in M_{co}^1(Cl)$$

Аналогічно контекстом рівня k буде

$$Con_{kn}^k(Cl_{co}) = \{M_{co}^k(Cl), M_{co}^k(LnCl)\}$$

$$\text{де } M_{co}^k(Cl) = M_{co}^{k-1}(Cl) \cup M_{co}^k(Cl)'$$

$$\forall Cl_i \in M_{co}^k(Cl) \exists (LnCl_j, Cl_l \in M_{co}^{k-1}) : Cl_i \in SqLnCl_j, Cl_l \in SqLnCl_j$$

$$\forall LnCl_i \in M_{co}^k(LnCl) : \forall Cl_j \in SqLnCl_i : Cl_j \in M_{co}^k(Cl)$$

Контекстом певного класу Cl_{co} будемо вважати контекст максимального рівня m який можна побудувати у даній локальній базі знань.

$$Con_{kn}(Cl_{co}) = Con_{kn}^m(Cl_{co}) \perp m = \max(k) \tag{2}$$

З використанням формули (2) визначимо контекст моделі Md_{co} на рівні знань як контекст усіх класів, які можуть ініціалізувати слоти цієї моделі ($RgSlScMdco$).

$$Con_{kn}(Md_{co}) = \bigcup Con_{kn}(Cl_{co}) \perp Cl_{co} \in RgSlScMd_{co}$$

Джерела контекстних даних

Для вирішення поставленої задачі моделлю часто необхідно отримати додаткову інформацію з контексту визначених об'єктів. Ця інформація може зберігатися як в локальній базі знань, так і у зовнішніх джерелах. Доцільно допустити існування багатьох різнотипних джерел контекстної інформації, що відрізняються своїми власниками, спеціалізацією, форматами збереження даних та ін. Визначимо такі джерела контекстної інформації (рис 3):

- локальна база фактів. Отримання потрібних фактів з локальної бази здійснюється шляхом прямого запиту до контексту визначеного об'єкта
- зовнішні бази знань, що використовують ту ж онтологію, що й локальна база знань. Отримання потрібних фактів також здійснюється шляхом прямого звертання до контексту об'єкта
- зовнішні бази знань, які використовують інші онтології. При звертанні до цих джерел необхідно реалізувати відображення класів онтології як при формуванні запиту, так і при інтерпретації результату. Аналогічна операція відображення проводиться і при звертанні до структурованих джерел, що не використовують онтології (наприклад, реляційних баз даних)
- неструктуровані джерела інформації, наприклад веб. При звертанні до цих джерел помічними є технології, які розробляються в межах семантичного веб, зокрема семантична анотація веб-ресурсів [6]

• сервіси контекстних даних (context widgets) [5]. Сервіси контекстних даних надають інформацію по результатам моніторингу заданих параметрів системи. Взаємодія з такими сервісами виконується відповідно до визначених та опублікованих сервісами інтерфейсів.

Опрацювання контекстної інформації у когнітивній системі відбувається як підчас розумування та прийняття рішення з використанням моделей, так і підчас навчання системи або видобування знань.

Опрацювання контексту у процесі прийняття рішень

Розглянемо способи взаємодії з контекстом які відбуваються підчас взаємодії моделей. Інтелектуальна система побудована з використанням моделей досягає своєї мети шляхом активації та взаємодії моделей. Якщо окрема модель є відносно простою, то комплекс активованих моделей здатний вирішувати складні задачі.

Моделі взаємодіють тоді, коли модель, що виконується в даний момент потребує вирішення певної проміжної задачі, реалізованої в інших моделях. Біжуча модель стає моделлю – активатором, а обрана для вирішення нової задачі модель – активованою моделлю. Вибір активованої моделі здійснюється окремою моделлю – зв'язком моделей, яка вирішує задачі релевантності та оптимальності вибору.

Важливою особливістю моделі, яка відображає можливість її багаторазового використання, є те, що слоти моделі можуть ініціалізуватися об'єктами різних класів. Таким чином слотам моделі відповідають семантичні класи онтології які є Ролями. Наприклад, класи *Покупець*, *Продавець*, *Гроші*, *Товар*, *Клієнт* та ін. Відігравати ту чи іншу роль можуть об'єкти різних

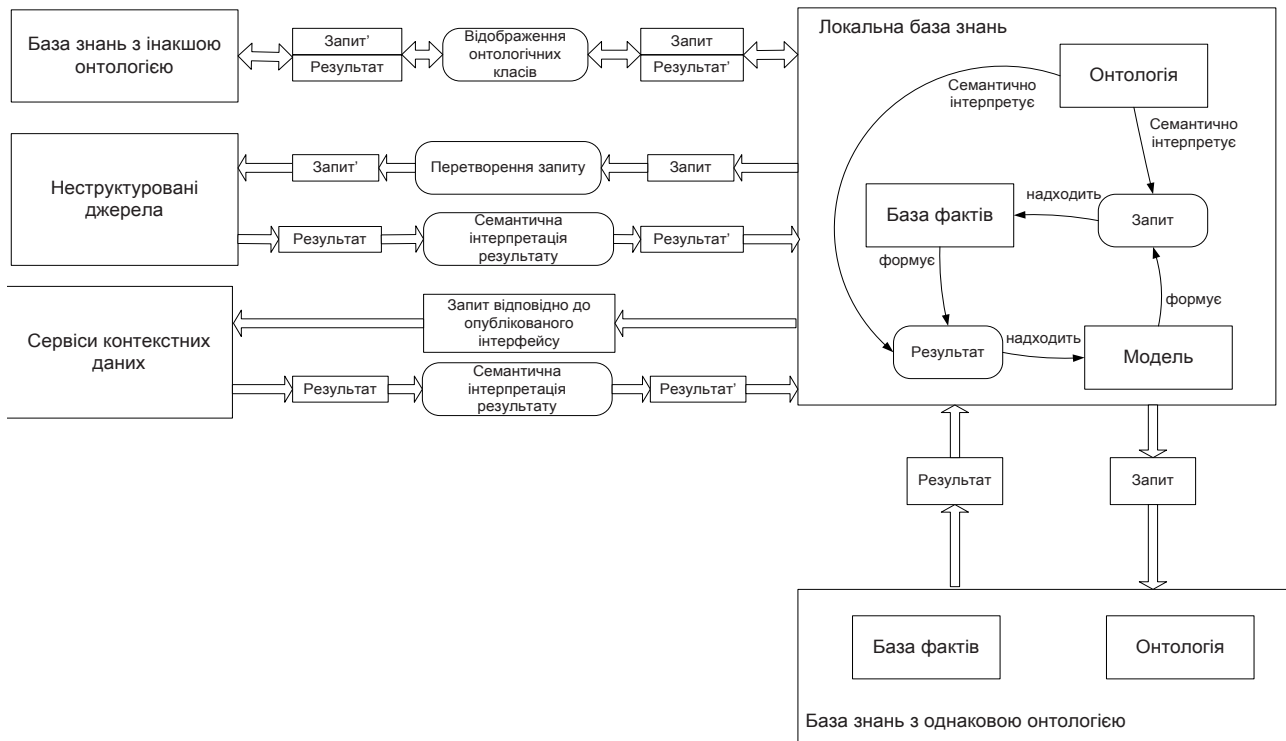


Рис. 3. Джерела контекстних даних

класів, наприклад *Покупцем* можуть бути як *Особа*, так і *Організація*.

Підчас встановлення зв'язку моделей виникає досить складна проблема відображення концептів які використовує активатор у концепти активованої моделі, адже ці моделі оперують у загальному випадку різними концептами. Модель-активатор повинна бути структурно незалежною від активованої моделі – цим досягається гнучкість та можливість використання різноманітних моделей для вирішення подібних задач. З іншого боку, активована модель повинна отримати від активатора семантично інтерпретовані дані необхідні для вирішення задачі.

Вирішити цю проблему пропонується заданням базової моделі яка є частиною зв'язку моделей. Базова модель призначена для формулювання постановки задачі і для вирішення широкого класу подібних задач. Фактично, така модель використовується багатьма активаторами для вирішення подібних задач. Базова модель містить мінімальний набір концептів та залежностей, необхідний для вирішення задачі активованою моделлю та який відповідає концептам цієї моделі. Передбачається, що активована модель, яка має ширший набір концептів, зможе отримати усю недостатню інформацію з контексту. Таким чином, базова модель виступає частиною зв'язку моделей і частиною протоколу взаємодії моделей. Вона містить формальну постановку задачі, яку потрібно вирішити і не містить опису процесу вирішення.

Позначимо модель-активатор Md_{ac} , базову модель Md_b . Тоді операція ініціалізації яка відбувається під час активації зв'язку моделей можна подати як:

$$InLnMd_{ac} : Con(Md_{ac}) \rightarrow M(ValSMd_b)$$

Операцію ініціалізації можна подати як формулювання постановки задачі, яку потрібно вирішити з використанням активованої моделі.

Нехай множина моделей, які можуть бути активовані для вирішення задачі, описаної базовою моделлю $Md_b \in M(Md_{at})$. Тоді існує однозначне відображення між слотами базової моделі та слотами активованих моделей. Таке відображення найпростіше провести за типами компонент базової та активованої моделей, так що для кожного слоту базової моделі SMd існує тільки один слот активованої моделі, які належать одному типу.

$$\forall(Md_b, Md_{at}) : \forall SMd_b (\exists^1 SMd_{at} \wedge Type(SMd_b) = Type(SMd_{at}))$$

Таким чином, множина типів слотів базової моделі є підмножиною типів слотів кожної допустимої для активації моделі.

$$M(Type(SMd_b)) \subseteq M(Type(SMd_{at})) \quad (3)$$

Операцію відображення значень слотів базової моделі на значення слотів активованої моделі будемо вважати першою стадією ініціалізації активованої моделі - $InLnMd_{at}^1$

Після цього активована модель має посилання на об'єкти бази знань з моделі-активатора інтерпретовані ролями базової та активованої моделей. Ці посилання використовуються активованою моделлю для доступу

до контексту зазначених об'єктів та отримання додаткових даних необхідних для вирішення поставленої задачі. Це відбувається на другій стадії ініціалізації активованої моделі:

$$InLnMd_{at}^2 : Con(Md_{ac}) \rightarrow M(ValSMd_{at})$$

Взаємодія з сервісами контекстних даних у контексті активної моделі

Важливою складовою частиною систем, що працюють з контекстом є сервіси контекстних даних [5]. Прикладами таких сервісів є різноманітні сенсори, системи моніторингу параметрів, агенти та компоненти, що збирають визначені дані, веб-сервіси та ін. Сервіси контекстних даних, залежно від реалізації бувають як інтелектуальними, так і неінтелектуальними.

Багато моделей, а особливо такі, що мають завдання розпізнавання та інтерпретацію подій та тенденцій навколишнього середовища, використовують у своїй роботі дані, добуті сервісами контекстних даних.

Аналогічні за змістом завдання виконують різноманітні системи моніторингу. На сьогодні системи моніторингу відслідковують достатньо вузьке коло параметрів та записують результати у текстові файли реєстрації або в реляційну базу даних. В обох цих випадках відсутня семантична інтерпретація результату моніторингу. Аналіз даних моніторингу виконується людиною вручну або за допомогою комп'ютерної програми, яка форматує, фільтрує дані. Такий підхід має ряд суттєвих недоліків, серед яких відсутність гнучкості, обмежений набір параметрів моніторингу, що у свою чергу пояснюється обмеженістю ресурсів, неефективне використання системних ресурсів на моніторинг якщо його результати не використовуються, ручний аналіз результатів тощо.

Підвищити ефективність моніторингу параметрів середовища в когнітивній системі можна шляхом створення завдань на моніторинг у контексті моделей які виконуються. Це дозволить:

- ефективніше використовувати системні ресурси – збиратиметься інформація тільки про параметри, визначені моделями що виконуються у даний момент часу, у контексті цих моделей
- надати системі моніторингу гнучкості. Якщо модель виявить певну важливу конфігурацію подій, вона може сформулювати нове завдання на моніторинг, яке надасть дані для поглибленого аналізу та прийняття рішення
- проводити аналіз результатів моніторингу та прийняття рішення інтелектуальною системою, а тому виконати їх оперативніше та швидше, ніж аналогічні дії проведені людиною.

Взаємодія моделі з сервісом контекстних даних відрізняється від її взаємодії з іншою моделлю. Зокрема, якщо при взаємодії моделей активатор очікує на отримання результату від активованої моделі, то підчас взаємодії з сервісами контекстних даних модель-активатор формулює завдання на моніторинг, яке враховує як потреби моделі, так і можливості сервісу. Результати моніторингу накопичуються у базі знань протягом значного часу і використовуються при наступних активаціях моделі.

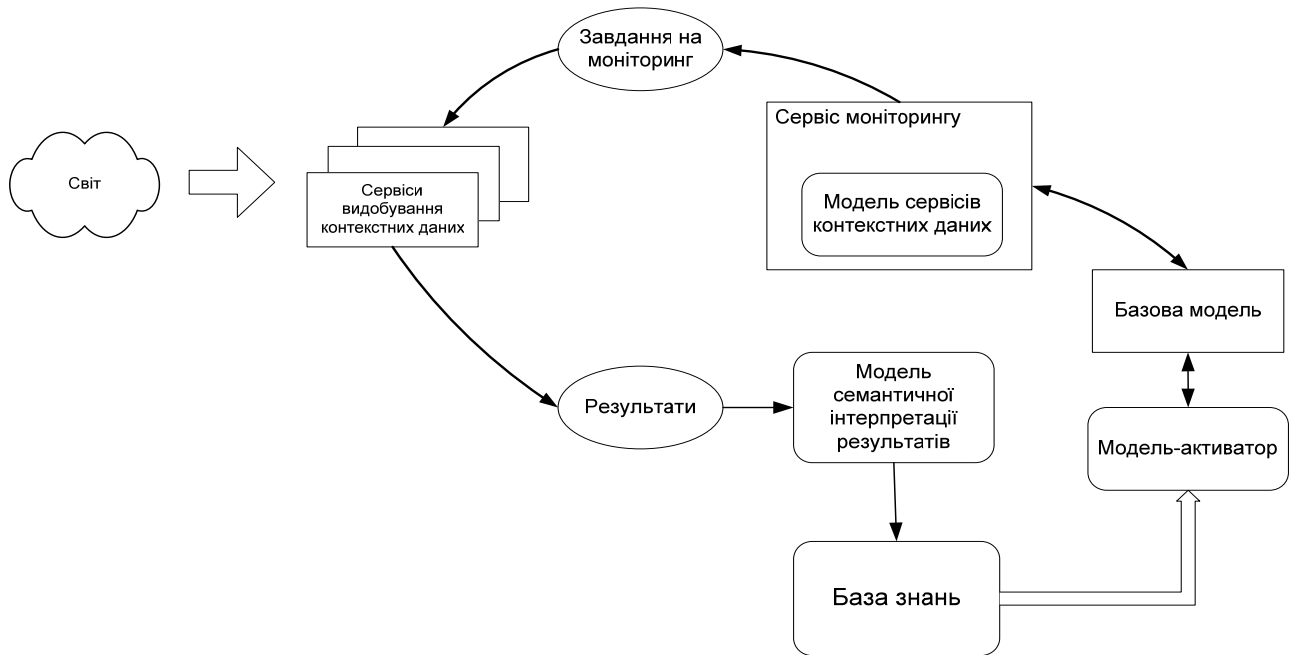


Рис. 4. Процес взаємодії моделі активатора з сервісами контекстних даних

Процес взаємодії моделі активатора з сервісами контекстних даних зображено на рис. 4. Модель – активатор формулює свої потреби у контекстних даних у вигляді базової моделі.

Базова модель опрацюється сервісом моніторингу, що містить модель сервісів контекстних даних. Цей сервіс є окремим семантично-орієнтованим застосуванням, керованим моделями. Він має власну локальну базу знань. Сервіс моніторингу володіє інформацією про доступні сервіси контекстних даних та їх можливості та обмеження, протоколи взаємодії з цими сервісами, завдання на моніторинг які вже виконуються, ресурсні обмеження, способи проведення моніторингу та ін. Аналізуючи запит на моніторинг, що надійшов від моделі –активатора, сервіс моніторингу приймає рішення щодо можливості виконання запиту активатора, обирає відповідний сервіс контекстних даних, формулює завдання на моніторинг відповідно до базової моделі та вимог сервісу, інформує активатор про прийнятий (або відхилений) запит. Завдання на моніторинг по формі відповідає вимогам інтерфейсу сервісу контекстних даних, а за змістом відповідає запиту моделі-активатора. При необхідності отримання додаткових даних сервіс моніторингу звертається до контексту моделі активатора.

Сервіси видобування контекстних даних формують результати моніторингу. При необхідності ці результати додатково семантично інтерпретуються перед записом у базу фактів моделлю семантичної інтерпретації.

Опрацювання контексту у процесі навчання

Однією з основних вимог до когнітивної системи є здатність самостійно (або за допомогою вчителя) поповнювати знання, навчатися. Під навчанням ми будемо розуміти процес взаємодії з зовнішніми джерелами знань (базами знань, вчителем, експертними система-

ми) в результаті якого локальна база знань поповнюється новими знаннями.

Ми будемо розглядати один з можливих різновидів навчання, а саме навчання у процесі вирішення реальних завдань, яке може бути віднесено до відомого підходу навчання та розумування на прикладах (Case based learning and reasoning) [7].

Процес навчання має такі особливості:

навчання завжди відбувається в ході вирішення певної задачі, реалізації конкретної моделі, в контексті цієї моделі. В результаті навчання поповнюється контекст цієї моделі на рівні знань.

в результаті навчання змінюються як декларативні (онтології) так і процедурні (моделі) знання, додаються нові класи онтології, зв'язки, правила та обмеження, нові моделі та посилання на них.

Головні об'єкти задіяні у процесі навчання подані на рис. 5

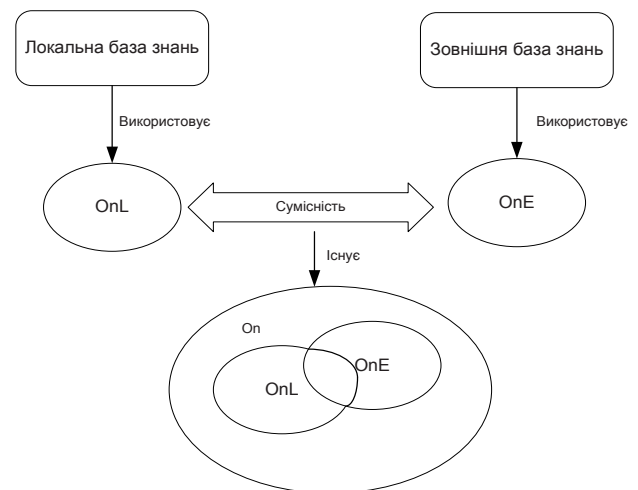


Рис. 5. Процес навчання з використанням сумісних онтологій

Нехай локальна база знань використовує онтологію OnL, а зовнішня – OnE. Важливим обмеженням, необхідним для реалізації автоматичного навчання є сумісність онтологій OnL, OnE. Будемо вважати пару онтологій {OnL, OnE} сумісною якщо існує (або може бути побудована) онтологія On, така що як OnL, так і OnE є частинами цієї онтології.

$$\exists On : OnE \subseteq On \wedge OnL \subseteq On$$

Якщо {OnL, OnE} несумісні, то необхідно провести узгодження (відображення) онтологій, що є достатньо складною операцією і зазвичай проводиться вручну [8].

Нехай у процесі функціонування системи біжуча активна модель визначає, що для вирішення своєї задачі необхідно вирішити певну підзадачу. Вона формулює цю підзадачу у вигляді базової моделі. Якщо в локальній базі знань відсутні моделі, які дозволяють вирішити поставлену підзадачу або наявні моделі не задовільняють сформульовані у базовій моделі обмеженням, то необхідно звернутися до зовнішніх баз знань у пошуку потрібної моделі.

Пошук моделі проводиться з використанням окремої онтології цілей (призначень) моделей OnGl: $On_{Gl} \subseteq On$ Вона є частиною загальної спільної онтології On та забезпечує спільне розуміння цілей та призначень моделей. В результаті пошуку формується множина моделей, мета яких відповідає меті базової моделі Gl_b .

$$SearchMd(Gl_b) = M(Md_{se}) \perp \forall Md_i \in M(Md_{se}) : Type(Goal(Md_i)) = Type(Gl_b)$$

де Goal(Md) – функція що повертає мету моделі Md.

Співпадіння класів мети знайдених моделей з класом мети базової моделі забезпечує відповідність між базовою моделлю та знайденими моделями аналогічно до формули (3).

$$M(Type(SlMd_b)) \subseteq M(Type(SlMd_{se}))$$

З отриманої множини моделей виділяють підмножину релевантних моделей:

$$M(Md_{re}) \subseteq M(Md_{se})$$

для яких істинна функція релевантності визначена у контексті базової моделі:

$$\forall Md_{re} \in M(Md_{re}) : Re(Con(Md_b), Md_{re}) = true$$

Якщо множина релевантних моделей має декілька елементів, то до них застосовують функцію вибору ChMd, яка визначає одну модель Md_{op} , яка є оптимальною, відносно до заданих у базовій моделі критеріїв вибору ChCrMdb.

$$Md_{op} = ChMd_b(Con(Md_b), ChCrMdb, M(Md_{re}))$$

На наступному етапі проводять поповнення локальної онтології OnL у контексті визначеної моделі Md_{op} згідно такого алгоритму.

1. Для кожного слоту SlScMd_{op} визначають відповідний клас онтології OnE:

$$Cl'_{slmd} = Type(SlCsMd_{op})$$

2. Якщо це клас не належить онтології OnL ($Cl'_{slmd} \notin OnL$), то цей клас додають до онтології OnL:

$$OnL = OnL \cup Cl'_{slmd}$$

3. Інакше, якщо клас належить онтології OnL, то проводять узгодження визначень класів $Cl_{slmd} \in OnL$, $Cl'_{slmd} \in OnE$ так що для кожного слоту класу $SlCl'_{slmd} \notin Cl_{slmd}$ додають цей слот до класу: $Cl_{slmd} = Cl_{slmd} \cup SlCl'_{slmd}$

4. Аналогічно додають до класу Cl_{slmd} усі обмеження, правила, операції з Cl'_{slmd} яких бракує Cl_{slmd} в:

$$Cl_{slmd} = Cl_{slmd} \cup M(RuCl'_{slmd}) \perp \forall RuCl'_{slmd} \notin Cl_{slmd}$$

$$Cl_{slmd} = Cl_{slmd} \cup M(CsCl'_{slmd}) \perp \forall CsCl'_{slmd} \notin Cl_{slmd}$$

$$Cl_{slmd} = Cl_{slmd} \cup M(OpCl'_{slmd}) \perp \forall OpCl'_{slmd} \notin Cl_{slmd}$$

Після узгодження онтологій відбувається імпорт моделі Md_{op} до локальної бази знань, встановлення посилаць на цю модель в інших моделях Md' локальної бази знань для яких

$$Goal(Md') = Goal(Md_{op})$$

Видобування знань з контексту

Видобування знань у семантично-орієнтованих системах суттєво відрізняється від видобування знань в інших системах. Головною відмінністю є то, що основою для аналізу даних виступають факти з бази знань – семантично інтерпретовані факти. У процесі аналізу даних використовують наявні у базі знань властивості, залежності та обмеження (існуючі знання). Результатом видобування знань є нові знання, які отримують у формі постійних значущих конфігурацій. На основі цих конфігурацій формують нові концепти, обмеження, моделі і правила.

Поняття контексту відіграє важливу роль у процесі видобування знань у семантично-орієнтованих системах. По-перше, процес видобування знань завжди відбувається з певною метою і таким чином, у контексті цієї мети. По-друге, контекст мети визначає і обмежує набір фактів, які аналізують у процесі видобування знань. Все це спрощує вирішення поставленої задачі.

Часто видобування знань проводять у контексті визначеного факту (набору фактів) або контексті певної системи. У цьому випадку будемо говорити про процес видобування знань з контексту (Context mining).

У процесі видобування нових знань використовуються моделі пошуку знань, які інкапсулюють підходи до пошуку знань, які застосовуються у практиці формування нових знань людиною. Наприклад, якщо треба розробити модель для досягнення визначеної мети Gl, система виділяє з бази знань стани, в яких ця мета була досягнута та досліджує дії, що передували досягненню цієї мети. В результаті порівняння багатьох послідовностей дій відсіюються несуттєві дії та виділяються дії, які наявні у всіх досліджуваних випадках. На основі цього формується модель – гіпотеза, що описує послідовність дій, які призводять до

визначеної мети. Гіпотеза додатково валідується, наприклад, шляхом проведення експериментів.

Видобування знань у семантично-орієнтованих системах, яке нерозривно пов'язане з застосуванням контекстних обмежень, є новим та перспективним напрямком наукових досліджень, що потребує вирішення багатьох наукових задач.

Висновок

Запроновані у роботі підходи до визначення та опрацювання контексту у когнітивній інформаційній системі дозволить створювати системи, які гнучко реагують на зміни вимог у бізнес – середовищі, адаптивні до змін, підтримують багатоваріантність у вирішенні задач.

Література

1. Буров Є.В. Опрацювання знань у когнітивній інформаційній системі керованій моделями [Текст]/ Буров Є.В.//Східно-Європейський журнал передових технологій, 2009.-№6/7(42).-с.40-49.

2. Context. [Electronic resource]. - Mode of access: WWW. URI:http://www.merriam-webster.com/dictionary/context.- Last access: 2009.- Title from the screen.
3. Dey A. Understanding and using context. [Text]/Dey A.//Journal of personal and ubiquitous computing, 2001. -pp 4-7.
4. Schmidt A. There is more to context than location./Schmidt A.,Beigl M, Gellersen H.//Computers & Graphics Journal.- Elsevier, 1999. -pp 893-902.
5. Raz D. Fast and efficient context-aware services [Text] /Raz D., Juhola A, Serrat-Fernandez J., Galis A.-John Wiley & Sons, Chichester, England.-2006.-222 p.
6. Daconta M. The Semantic web. A Guide To The Future of XML, Web-services and Knowledge Management /Michael C Daconta, Leo J Obrst, Kevin T Smith .- Wiley publishing.- 2003.- 305p.
7. Sankar Pal. Foundations of soft case-based reasoning/ Sankar Pal., Simon Shiu.- Johnwiley&sons,inc., 2004.- 299p.
8. Salam A.Semantic Web Technologies and E-Business:Toward the Integrated Virtual Organization and Business Process Automation./Salam A., Stevence J.- Idea Group Publishing, 2007.- 451 p.

Розглянуто динаміку ентропії й абсолютної організації вагових характеристик транспортних систем

Ключові слова: максимальна ентропія, текуча ентропія, абсолютна організація, оцінка станів

Рассмотрена динамика энтропии и абсолютной организации весовых характеристик транспортных систем

Ключевые слова: максимальная энтропия, текущая энтропия, абсолютная организация, оценка состояний

Dynamics of entropy and the absolute organisation of weight characteristics of transport systems is considered

Keywords: the maximum entropy, current entropy, the absolute organisation, an estimation of conditions

УДК 303.725.36

ДИНАМИКА ЭНТРОПИИ РАСЧЕТНЫХ ХАРАКТЕРИСТИК ТРАНСПОРТНЫХ СИСТЕМ

Н. В. Ярещенко

Кандидат технических наук, доцент
Кафедра транспортных технологий*

А. Г. Горелова*

*Национальный автомобильно-дорожный университет
ул. Петровского, 25, г. Харьков, Украина, 61002

1. Введение

Транспортная система, как и все технические системы, характеризуется чередованием периодов ускоренного роста с периодами относительно замедленных темпов. В периоды замедленных темпов роста осуществляется переход к новым методам исследования, к принципиально новым типам двигателей. В период

ускоренных темпов роста реализовались потенциальные возможности нововведений.

Для ускоренных темпов роста характерно замкнутое, в организационном отношении, состояние технической системы, а для замедленных – разомкнутое. Система замкнута, если между системой и средой вообще нет обмена веществом и энергией или, по крайней мере, такой обмен не приводит к увеличению макси-