

## МОЖЛИВОСТІ ЗАСТОСУВАННЯ ОНТОЛОГІЙ ПІД ЧАС СТВОРЕННЯ БАЗ ЗНАНЬ НАВЧАЛЬНИХ КОМП'ЮТЕРНИХ СИСТЕМ

Черемісіна Любов Олександрівна,

аспірантка кафедри теоретичних основ інформатики НПУ

ім. М. П. Драгоманова, [Lubami@ukr.net](mailto:Lubami@ukr.net).



**Анотація.** У статті розглядається застосування онтологій для використання та розробки навчальних комп'ютерних систем. Обґрунтовано актуальність створення інтелектуальних навчальних систем, заснованих на систематизованих знаннях. Розглянуто класифікацію властивостей, використання та переваг онтологій. Охарактеризовано підходи до вирішення проблеми відображення онтологій. Зроблено аналіз мов, призначених для формального опису онтологій. Розглянута проблема узгодженості онтологій та класифікація комп'ютерних інструментальних засобів, що «виявляють» відповідності між онтологіями. Охарактеризовано практичне значення розробки онтологій.

**Ключові слова:** штучний інтелект, база знань, онтологія, навчальні системи.

Обсяги інформації з кожним роком стрімко збільшуються, відбувається перевантаження інформацією. Потік інформації надходить з науки, бізнесу, Інтернету та інших джерел. Але знання є не лише у людини, вони містяться також у накопичених даних, які необхідно аналізувати. Такі знання часто називають «прихованими».

Очевидно, що для виявлення «прихованих» знань потрібно застосовувати спеціальні методи автоматичного аналізу даних, за допомогою яких доводиться добувати знання із величезного обсягу інформації.

Таким чином, актуальною є задача створення навчальних комп'ютерних систем (НКС), заснованих на систематизованих знаннях.

Ефективним засобом представлення та систематизації знань є онтології. Онтології використовуються для формальної специфікації понять і відношень, які характеризують певну галузь знань. Перевагою онтологій як способу представлення знань є їх формальна структура, яка спрощує комп'ютерну обробку інформації.

Можна виділити такі етапи практичного розроблення онтологій: визначення класів в онтології; розташування класів в таксономічну ієрархію (підклас — надклас); визначення слотів і опис значень цих слотів, що допускаються; заповнення значень слотів екземплярів.

Після цього можна створити базу знань, визначивши окремі екземпляри цих класів, ввівши в певний слот значення і додаткові обмеження для слота.

Забезпечення можливості використання знань предметної області стало однією з рушійних сил недавнього сплеску у вивченні онтологій. НКС, що використовують онтології, знаходять застосування у розробленні та моделюванні систем представлення й інтерпретації інформації в процесі навчання. Онтологія визначає терміни, за допомогою яких можна описати і структурувати предметну галузь. Використання онтологій ефективно під час пошуку й об'єднання інформації з різних джерел і середовищ. Мова онтологій використовується для надання інформації чітко визначеного значення і є загальним набором термінів для опису та подання предметної галузі, що вивчається.

Електронні освітні ресурси (ЕОР) активно розробляються й упроваджуються в навчальний процес у більшості країн з необхідним розвитком освіти і технологій [1, 6]. У створенні предметного наповнення для найрізноманітніших типів інформаційних систем (ІС) та ЕОР набуває все більшого використання онтологічний підхід [2–8]. Про зростаючу роль онтологій для ІС свідчить також уведення в роботу [16] поняття керованих онтологіями інформаційних систем (Ontology-Driven Information Systems — ODIS).

Онтології можуть використовуватися в ODIS під час їх розробки, використання, інтеграції, а також можуть підтримувати різні частини інформаційної системи — інтерфейси користувача, бази даних чи інші компоненти.

Утім, нині не існує загальноприйнятого визначення «онтології», різняться погляди і на те, що вони мають включати в себе [4, 9].

Щодо методів побудови онтологій, то на сьогодні існує набір мов, призначених для формального опису онтологій. Серед найбільш відомих і використовуваних: KIF (Knowledge Interchange Format), DAML+OIL (DARPA Agent Mark up Language), OWL (Ontology Web Language). Існують також інструментальні засоби, що підтримують розробку онтологій відповідно до цих специфікацій.

Аналізуючи конструкції даних мов формального опису, можна помітити, що навіть в найбільш розширених з них OWL, що включає OIL+DAML, існують детальні можливості лише для задавання класів, підкласів та їх членів (таксономії), для інших же типів відношень не передбачаються спеціальні елементи — їх можна задавати лише через властивості класів. На практиці більшість із вже створених онтологій є максимум ієрархічною структурою понять предметної галузі. Тобто розробниками розглядаються лише такі відношення між поняттями, як «вид-клас» та рідше «об'єкт-атрибут».

Тоді, як у деяких роботах [11], розроблено класифікацію властивостей онтологій та уточнено перелік структурних властивостей (таксономічні зв'язки, композиційні зв'язки, топологічні зв'язки, зв'язки сутностей з процесами, причинно-наслідкові зв'язки, часові та просторові зв'язки).

Відповідно до концепції Semantic Web створення RDF-описів та OWL-онтологій покладене на окремих розробників. І на сьогодні, попри досить незначні напрацювання в плані розробки онтологій, вже виникає проблема узгодженості онтологій, яка полягає в тому, що різними розробниками для однієї й тієї ж предметної галузі можуть бути створені онтології, синтаксично або семантично гетерогенні, і для їх сумісного використання необхідна трансляція або відображення (виявлення відповідності між поняттями двох онтологій) [12].

Існує проблема відмінності між онтологіями, яка полягає в тому, що якщо дві онтології були створені різними розробниками для однієї предметної галузі, то вони будуть суттєво відрізнятися і за таксономією, і за лексикою, і деякою мірою за семантикою. Семантична не-

однорідність онтологій є наслідком опису однакових онтологічних концептів різними термінами (синоніміка), призначення різних значень одному і тому ж концепту в різних контекстах (багатозначність). Неоднорідність в онтологіях може також викликатися різними таксономіями концептів.

З цієї причини задача об'єднання нетривіальна і в основному може розв'язуватися в напівавтоматичному режимі, тому що тільки експерт може остаточно підтверджувати коректність семантики встановлених відносин між онтологічними поняттями. Отже, існує велика потреба у високоефективних засобах автоматизації порівняння й інтеграції онтологій.

У задачах порівняння й об'єднання онтологій під час поетапного створення бази знань НКС існує чотири основні проблеми [2].

1. Семантичні відмінності: деякі онтологічні концепти описують однаковий домен різною термінологією (синонімія і багатозначність). Це призводить до появи доменів, що перетинаються. Таким чином, існує потреба в засобах картування, які змогли б інтерпретувати описи навчальних ресурсів, використовуючи лексичний та семантичний аналіз для того, щоб вирішити проблеми синонімії та багатозначності.

2. Структурні відмінності: різні розробники онтологій анотують цифрові навчальні ресурси різними онтологічними концептами. Це створює синтаксичні відмінності, вимагаючи застосування інструментальних засобів для об'єднання, які змогли б різні таксономії об'єднувати в загальну таксономію.

3. Масштабування операцій картування й об'єднання онтологій: це важливо особливо для великих онтологічних репозиторіїв.

4. Відсутність апріорних знань: апріорні знання необхідні для картування й об'єднання онтологій з використанням контрольованих методів продукування знань. Однак такі знання не завжди доступні, або їх надійність вимагає додаткового обґрунтування (у випадку прийняття за апріорну таксономії, визначеної методом експертного оцінювання).

Існує кілька підходів до розв'язання проблеми відображення онтологій, перший з яких — ручне відображення, шляхом встановлення відношень між концептами, підхід реалізовано для деяких великих онтологій [13]. Проблема застосування ручного відображення полягає в тому, що розмір онтологій може бути дуже великим і продовжуватиме нарощуватися, що вимагає надзвичайно багато людських зусиль для їх відображення. Тому, природно, що дослідники шукають шляхи відобразити онтології автоматично.

Досить значна кількість досліджень, серед яких і [14], присвячені розробці засобів відображення онтологій на основі методів машинного навчання, серед яких особливою популярністю користуються методи класифікації текстів. Однак результати тут залежать від якості навчальних даних, а підготовка їх вручну для соціальних понять досить складна і дорога, що зменшує привабливість текстової класифікації.

Ще одним підходом до відображення онтологій є зіставлення імен понять на основі їх лексичної подібності та використання спеціально розроблених словників (Word Net), у яких описані відношення між концептами (синонімія) та властивості низки концептів.

Альтернативним напрямком досліджень є автоматичне створення онтологій, яке на сьогодні, у більшості випадків, зводиться до автоматичного анотування текстів у Web. Аналіз робіт у цьому напрямку подано

в [15], де показано їх обмеження виділенням певного типу відношень для анотування, або ж використанням для анотування певної онтології. Результати аналізу пропонується автоматично відображати у OWL — описи за допомогою OntoSem2OWL.

Дослідники в різних галузях інформатики працюють над автоматичним об'єднанням онтологій. Однак, і автоматичне об'єднання онтологій і створення інструментальних засобів, знаходяться на стадіях розвитку.

Розглянемо деякі комп'ютерні інструментальні засоби, що «виявляють» відповідності між онтологіями. Їх можна класифікувати так [7]:

- Prompt, Chimaera, Onto Merge — об'єднують дві онтології з метою створення однієї нової;
- Onto Morph — визначає функції перетворення, що перетворює одну онтологію в іншу;
- GLUE, OBSERVER, FCA-Merge — визначають відображення між концептами у двох онтологіях, знаходячи пари зв'язаних концептів;
- ONION — визначає правила відображення для зв'язку тільки релевантних частин «початкових» онтологій.

Розглянемо тепер вищезгадані засоби більш детально.

**Chimaera** — інтерактивний інструмент для об'єднання, заснований на редакторі онтологій Ontolingua. Chimaera дозволяє користувачу з'єднувати онтології, розроблені в різних формалізмах. Користувач може запросити аналіз або керівництво від Chimaera у будь-якій точці протягом процесу об'єднання. Chimaera головним чином покладається на те, з якої онтології прибули концепти, засновуючись на їхніх іменах. Chimaera цілком залишає вирішення того, що робити користувачу і не робить жодних пропозицій самостійно. Єдине таксономічне відношення, що розглядає Chimaera, — відношення підклас-суперклас.

У **Onto Merge** об'єднана онтологія є об'єднанням двох початкових онтологій і набору аксіом сполучення. Перший крок у процесі об'єднання Onto Merge, полягає в трансляції обох онтологій до загального синтаксичного представлення мовою, розробленою авторами. Потім людина-експерт визначає аксіоми сполучення, що містять терміни з обох онтологій.

**Onto morph** визначає набір операторів перетворення, що можуть застосовуватися до онтологій. Потім людина-експерт використовує початковий список пар і початкові онтології для визначення набору операторів, що повинні застосовуватися до початкових онтологій для усунення розходжень між ними, і Onto morph застосовує ці оператори. Таким чином, сукупні операції можуть бути виконані за один крок. Однак, експерт не одержує ніякого керівництва за винятком початкового списку співпадаючих пар.

**GLUE** — система використовує методи машинного навчання для знаходження відображення. GLUE використовує інформацію в екземплярах концепту і таксономічній структурі онтологій. Для об'єднання результатів використовується вірогіднісна модель.

Система **OBSERVER** використовує дескриптивну логіку для відповіді на запити, що використовують декілька онтологій та інформацію щодо відображень між ними. Спочатку користувачі визначають набір міжонтологічних відношень. Система допомагає користувачам впоратися з цією задачею за допомогою пошуку синонімів у початкових онтологіях. Визначаючи відображення, користувачі можуть формулювати запити в термінах дескриптивної логіки, через свою власну онтологію. Потім **OBSERVER** використовує інформацію відображення для формулювання запитів до початкових онтологій. **OBSERVER** значною мірою покладається на той факт, що описи в онтологіях і запитах є змістовними.

**FCA-Merge** — метод для порівняння онтологій, що мають набір спільних екземплярів або набір спільних документів, що анотуються за допомогою концептів з початкових



онтологій. Ґрунтуючись на цій інформації, FCA-Merge використовує математичні методи з Formal Concept Analysis для того, щоб зробити решітку концептів, що зв'язує концепти з початкових онтологій. Алгоритм пропонує відношення еквівалентності і підклас-суперклас. Потім інженер може аналізувати результат і використовувати його як підказки для створення об'єднаної онтології. Однак, припущення, що дві поєднувальні онтології використовують спільний набір екземплярів або мають набір документів, у якому кожен документ анується термінами обох джерел занадто перебільшене, і на практиці така ситуація відбувається рідко.

Система **ONION** заснована на алгебрі онтологій. **ONION** використовує лексичні методи та методи на основі графів для пропозиції артикуляції. Метод знаходить лексичну подібність між іменами концептів, використовуючи словники і методи семантичної індексації, засновані на спільному місцезнаходженні групи слів у текстовій сукупності.

**PROMPT** — доповнення до системи Protege, є алгоритмом для об'єднання і групування онтологій. Об'єднуючи дві онтології, **PROMPT** створює список запропонованих операцій. Операція може полягати в об'єднанні двох термінів або копіюванні термінів у нову онтологію. Користувач може виконати операцію вибором однієї із запропонованих або безпосереднім визначенням операції. **PROMPT** виконує обрану операцію і додаткові зміни, що викликає ця операція. Потім список запропонованих операцій модифікується і створюється список конфліктів і можливих вирішень цих конфліктів. Це повторюється поки не буде готова нова онтологія.

Розмаїтість у типах інструментальних засобів робить складним їх безпосереднє порівняння. Фактично, коли розроблювач повинен вибрати інструмент для використання в керуванні множинними онтологіями, який комп'ютерний засіб є найбільш придатним буде залежати від конкретної задачі. Наприклад, якщо поєднувані онтології спільно використовують набір екземплярів, то найкраще може працювати FCA-Merge. Якщо тільки частини онтологій повинні відображатися, можна було б вибрати інструмент **ONION**. Якщо онтології мають дуже обмежену структуру, а концепти мають докладні визначення однаковою природною мовою, інструментальні засоби ISI/USC можуть забезпечувати кращі відповіді. Якщо екземпляри взагалі не доступні, і онтології містять багато відношень між концептами, **Prompt** може працювати найкраще.

Завдання об'єднання онтологій є найбільш часто вирішуваним при створенні бази знань НС. Оскільки традиційний підхід до навчання в наш час суттєво потіснили альтернативні методи навчання, які базуються на інформаційних технологіях. Сучасні НКС є засобом представлення інформації, засвоєння знань і умінь, проміжної і підсумкової перевірки. Онтологічний підхід до створення предметного наповнення ЕОР може бути досить продуктивним, оскільки за його застосування до змісту навчання кількох предметів можна отримати абсолютно нове бачення міжпредметних зв'язків [10].

★ ★ ★

**Черемисина Л.А. Возможности применения онтологий при создании баз знаний учебных компьютерных систем**

**Аннотация.** В статье рассматривается применение онтологий для использования и разработки учебных компьютерных систем. Обоснована актуальность создания интеллектуальных обучающих систем, основанных на систематизированных знаниях. Рассмотрено преимущества использования и классификацию свойств онтологий. Охарактеризованы подходы к решению проблемы отражения онтологий. Сделан анализ языков, предназначенных для формального описания онтологий. Рассмотрена проблема согласованности онтологий и классификация компьюте-

рных инструментальных средств, которые «обнаруживают» соответствия между онтологиями. Охарактеризованы практические значение разработок онтологий.

**Ключевые слова:** искусственный интеллект, база знаний, онтология, интеллектуальные системы.

★ ★ ★

**Cheremisina L. Possibilities ontology use when creating knowledge bases educational computer systems**

**Annotation.** This paper deals with the use of ontology for the use and development of tutoring systems. Actuality creating intelligent tutoring systems based on systematic knowledge. We consider classification of properties, use and benefits of ontology. The characteristic approach to the problem of mapping ontology. The analysis of language intended for the formal description of ontology. Considered problem of ontology consistency. We consider the classification of programs that «show» the correspondence between anthologies. The characteristic of practical importance anthologies.

**Key words:** artificial intelligence, knowledge base, ontology, intelligent systems.

## Література

1. Биков В. Ю., Лапінський В. В. Методологічні та методичні основи створення і використання електронних засобів навчального призначення // Комп'ютер в школі і сім'ї — 2010. — №2 (98). — С. 3–6.
2. Вороной О. С., Ерошина Г. А. Засоби інтеграції онтологій предметних областей для створення баз знань інтелектуальних навчальних систем // Штучний інтелект. — 2010. — №2. — С. 124–130.
3. Келеберда И. Н., Лесная Н. С., Репка В. Б. Использование мультиагентного онтологического подхода к созданию распределенных систем дистанционного обучения // Educational Technology & Society. — 2004. — №7 (2).
4. Клещев А. С., Артемьева И. Л. Математические модели онтологий предметных областей. Часть 1. Существующие подходы к определению понятия «онтология» / Научно-техническая информация. — Серия 2 «Информационные системы и процессы». — 2001. — №2. — С. 20–27.
5. Козак І. А. Централізований підхід до опису Web-онтологій // Штучний інтелект. — 2008. — №4. — С. 80–85.
6. Лапінський В. В., Міна А. С., Скрипка К. І. Міжнародні тенденції розвитку інформатизації освіти та підвищення її якості // Інформаційні технології і засоби навчання — 2010. — №5 (19). — Режим доступу: <http://lib.iitta.gov.ua/230/>.
7. Овдій О. М., Проскудіна Г. Ю. Онтології у контексті інтеграції інформації: представлення, методи та інструменти побудови // Пробл. программирования. — 2004. — №2–3. — С. 353–365.
8. Палагин А. В., Величко В. Ю., Стрижак А.Е., Попова М. А. Инструменты поддержки процессов аналитической деятельности эксперта при тематическом исследовании информационных ресурсов и источников // International Journal «Information Technologies and Knowledge» Vol. 4, Number 4, 2010. С. 329–347.
9. Пальчунов Д. Е. Моделирование мышления и формализация рефлексии I. Теоретико-модельная формализация онтологий и рефлексии // Философия науки. — 2006. — №4 (31). — С. 86–114.
10. Стрижак О. Е. Комп'ютерні тезауруси як технологічна платформа створення авторських методик викладання предметних дисциплін / О. Е. Стрижак // Актуальні проблеми психології: Психологічна теорія і технологія навчання / за ред. С. Д. Максименка, М. Л. Смольсон. — К.: Вид-во НПУ імені М. П. Драгоманова, 2009. Т.8. — Вип.6. — С. 259–266.
11. Шалфеева Е.А. Классификация структурных свойств онтологий // Искусственный интеллект. — 2005. — №3. — С. 67–77.
12. Dou D., McDermott D. Qi. P. Ontology Translation on the Semantic Web // In Proceedings of International Conference on Ontologies, Databases and Applications of Semantics (ODBASE 2003). LNCS 2888. — Berlin Heidelberg. Springer-Verlag. — 2003. — P. 952–969.
13. Niles I., Pease A. Mapping WordNet to the SUMO Ontology // Proc of the IEEE International Knowledge Engineering conference (2003).
14. Zhongli Ding, Yun Peng, Rong Pan, Yang Yu. A Bayesian Methodology towards Automatic Ontology // Proceedings of the AAAI-05 C&O Workshop on Contexts and Ontologies: Theory, Practice and Applications. — July 09, 2005.
15. Akshay Java, Sergei Nirneburg, Marjorie McShane, Timothy Finin, Jesse English, Anupam Joshi. Using a Natural Language Understanding System to Generate Semantic Web Content // Final version to appear, International Journal on Semantic Web and Information Systems. — 2007. — 3 (4). — Режим доступу: <http://www.igi-pub.com/>.
16. Guarino, N. Formal Ontology and Information Systems, in N. Guarino (Ed.) Formal Ontology in Information Systems. — Amsterdam, Netherlands: IOS Press (1998). — P. 3–15.