Извлечение семантических отношений из статей Википедии с помощью алгоритмов ближайших соседей

А.И. Панченко[2,1], С.А. Адейкин[1], А.В. Романов[1] и П.В. Романов[1]

[1] МГТУ им. Н.Э. Баумана, Системы Обработки Информации и Управления

[2] Catholic Univeristy of Louvain, Center for Natural Language Processing

17 марта 2012 г.

Plan

Семантические отношения

В рамках данной работы под семантическими отношениями понимаются:

- **синонимы** (отношения эквивалентности):
  $\langle car, SYN, vehicle \rangle, \langle animal, SYN, beast \rangle$
- **гиперонимы** (иерархические отношения):
  $\langle car, HYPER, Jeep\ Cherokee \rangle, \langle animal, HYPER, crocodile \rangle$
- **ко-гиперонимы** (общий гипероним):
  $\langle Toyota\ Land\ Cruiser, COHYPER, Jeep\ Cherokee \rangle$

Формально:

- $r = \langle c_i, t, c_j \rangle$ – **семантическое отношение**, где $c_i, c_j \in C$ – **слова**, такие как *radio* или *receiver operating characteristic*, $t \in T$ – **тип семантического отношения**, такой как *синонимия* или *гипонимия*
- $R \subseteq C \times T \times C$ – множество **семантических отношений**
- $R \subseteq C \times C$ – множество **нетипизированных отношений**

Семантические отношения фиксируются в ...

Тезаурусах: граф $G = (C, R)$

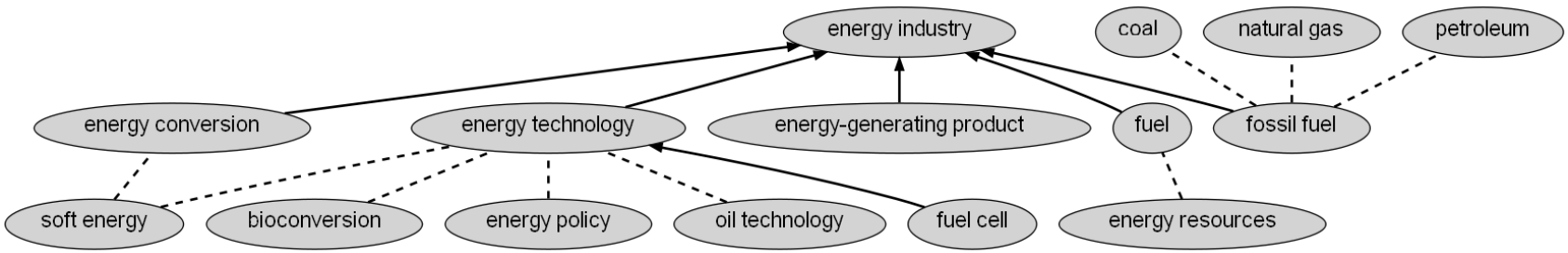


Рис.: Часть информационно-поискового тезуаруса EuroVoc.

$T = \{NT, RT, USE\}$
$R =$

- $\langle$energy-generating product, NT, energy industry$\rangle$
- $\langle$energy technology, NT, energy industry$\rangle$
- $\langle$petrolium, RT, fossil fuel$\rangle$

Других семантических ресурсах: онтологии, семантические сети, словари синонимов, терминологические классификаторы и т.п.

Применение семантических отношений

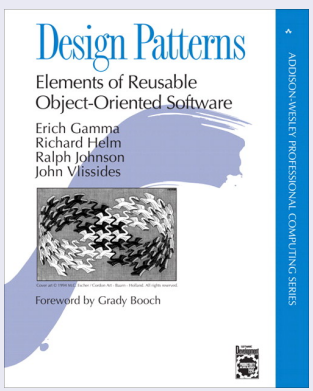
Семантические отношения представляют **знание о языке** полезное для различных приложений **автоматической обработки текста** (АОТ):

- Расширение и рекомендация поискового запроса в ИПС (Hsu et al., 2006)
- Построение вопросно-ответных систем (Sun et al., 2005)
- Категоризация текстовых документов (Tikk et al, 2003)
- Разрешение омонимии (Patwardhan et al., 2003)

Проблема

- Существующие ресурсы часто **недоступны или недостаточны** для
  - конкретного приложения
  - предметной области
  - языка

Пример: магазин продающий книги



“Design Patterns: Elements of Reusable Object-Oriented Software” $\Leftrightarrow$ “Gang of Four Book” $\Leftrightarrow$ GOF

- Как выдать в результате поиска книгу по запросу “GOF”?

Проблема

- **Ручное создание требуемых семантических ресурсов**:
  - (+) Точный результат
  - (–) Крайне дорогостоящий и трудоемкий процесс
  - (–) Неприменимо в большом количестве случаев
- Существующие методы извлечение отношений:
  - (–) Не обеспечивают достаточной точности
- Поэтому, актуальной задачей является разработка методов автоматического извлечения семантических отношений:

Состояние исследований и разработок

Существующие **методы** извлечения отношений . . .

- **лексико-синтаксические шаблоны** (Snow, 2004)
  - (+) точность, но малое покрытие
  - (–) ручное написание правил извлечения
  - (–) правила зависят от языка и предметной области
- **дистрибутивный анализ** (Филиппович и Прохоров, 2002; Grefenstette, 1994; Curran and Moens, 2002)
  - (+) не требует ручной работы
  - (–) точность

Существующие **метрики** семантической близости основанные на Википедии (Strube and Ponzetto, 2006; Gabrilovich and Markovitch, 2007; Zesch, Muller, and Gurevych, 2008):

- (+) точность
- (+) основные предметные области и языки
- (+) постоянно пополняется пользователями
- (–) не были использованы для извлечения отношений

Новизна работы

- Методы извлечения семантических отношений на основе:
  - текстов статей Википедии
  - двух метрик семантической близости – Cos, Overlap
  - двух алгоритмов – KNN, MKNN
- Система Serelex, реализующая предложенные методы
  - открытый исходный код (LGPLv3)

Данные и их предварительныая обработка

Данные:

- множество определений D английских слов C
- определение $d \in D$ – первый параграф статьи Википедии, название которой – $c \in C$
- источник статей – DBPedia.org

Предварительная обработка:

- Морфологический анализ (TreeTagger)
- Удаление стоп-слов
- 327.167 определений (237 Мб)
- 775 определений для теста (824 Кб)

```
axiom; in#IN#in traditional#JJ#traditional logic#NN#logic ,#,#, an#DT#an
axiom#NN#axiom or#CC#or postulate#NN#postulate is#VBZ#be a#DT#a
...is#VBZ#be not#RB#not proved#VVN#prove ...
```

Алгоритмы извлечения семантических отношений

Метод извлечения семантических отношений

Вход:

- C – множество слов,
- D – множество определений для C
- k – количество ближайших соседей

Выход:

- R – множество пар семантически связных слов

Алгоритмы

- KNN – Метод ближайших соседей
- MKNN – Метод взаимных ближайших соседей

Метрики семантической близости

- Cos – Косинус угла между векторами определений
- Overlap – Количество общих лемм в определениях

Метрики семантической близости

Вычисляют меру подобия **смысла слов** $c_i, c_j \in C$ на основе подобия определений $d_i, d_j \in D$

Overlap – Количество общих лемм в определениях

- $similarity(c_i, c_j) = \frac{2|(d_i \cap d_j|}{|d_i|+|d_j|}$
- $|d_j|$ – количество слов в определении $d_j \in D$

Cos – Косинус угла между векторами определений

- $similarity(c_i, c_j) = \frac{\mathbf{f}_i \cdot \mathbf{f}_j}{||\mathbf{f}_i|| \cdot ||\mathbf{f}_j||}$
- f_{ik} – частота леммы c_k в определении d_i
- $\mathbf{f}_i = (f_{i1}, \ldots, f_{in})$

Алгоритмы KNN

```
R = ExtractRelations(C, D, k, isMKNN)
Input: C — слова, D - определения слов, k - количество ближайших соседей,
isMKNN - если true использовать алгоритм MKNN, иначе KNN
Output: R - множество семантических отношений <c_i,c_j> in C X C
    1. //Вычисление попарной близости между всеми словами C
    2. Rmatrix = void
    3. for i=0; i<count(C); i++ {
    4.         for j=i; j<count(C); j++ {
    5.                 // Вычисляем семантическую близость двух слов
    6.                 s_ij = similarity(D(i), D(j))
    7.                 // Сохраняем наиболее подобные слова
    8.                 if( count(Rmatrix(C(i))) < k || s_ij > min(Rmatrix(C(i)) )
    9.                         Rmatrix(C(i)).addOrReplaceMin(C(j))
    10.                }
    11.        }
    12.}
```

Алгоритм MKNN

```
R = ExtractRelations(C, D, k, isMKNN)
Input: C – слова, D - определения слов, k - количество ближайших соседей,
isMKNN - если true использовать алгоритм MKNN, иначе KNN
Output: R - множество семантических отношений <c_i,c_j> in C X C
1.  //Вычисление попарной близости между всеми словами C
2.  Rmatrix = void
3.  for i=0; i<count(C); i++ {
4.          for j=i; j<count(C); j++ {
5.                  // Вычисляем семантическую близость двух слов
6.                  s_ij = similarity(D(i), D(j))
7.                  // Сохраняем наиболее подобные слова
8.                  if( count(Rmatrix(C(i))) < k || s_ij > min(Rmatrix(C(i)) ){
9.                          Rmatrix(C(i)).addOrReplaceMin(C(j))
10.                 }
11.         }
12. }
13. // Вычисление семантических отношений
14. R = void
15. foreach c_i in Rmatrix {
16.         foreach c_j in Rmatrix(c_i) {
17.                 if(!isMKNN || Rmatrix(c_j) contains c_i){
18.                         R.add(<c_i, c_j>)
19.                 }
20.         }
21. }
22. return R
```

- Временная сложность – $O(|C|^2)$
- Пространственная сложность – $O(k|C|)$

Пример работы KNN и MKNN

Computer	Apple	Fruit	
-	0.4	0.0	Computer
-	-	0.8	Apple
-	-	-	Fruit

$k = 2$

k ближайших соседей:

- Computer – { **Apple**, Fruit }
- Apple – { **Fruit**, Computer }
- Fruit – { **Apple**, Computer }

Пары для KNN: <Computer, Apple>, <Apple, Fruit>

Пары для MKNN: <**Apple, Fruit**>

Программный комплекс Serlex

- `http://github.com/jgc128/serelex`
- Язык: C++
- Используемые библиотеки: STL, boost
- Кроссплатформенная: Windows/Linux, 32/64-bit
- Интерфейс: консольный
- Многопоточность: Не используется
- Лицензия: LGPLv3

Эмпирическая оценка производительности:

- 755 дефиниций - 3 секунды
- 327 168 дефиниций – 3 дня 3 часа 47 минут
- Конфигурация сервера: Linux 2.6.32-cs-kernel с процессором Intel® Xeon® CPU E5606@2.13GHz

Извлеченные отношения

Пример извлеченных отношений R между множеством из 775 слов (MKNN, k=2, Overlap):

$R = \{$
$\langle acacia, pine \rangle, \langle aircraft, rocket \rangle,$
$\langle alcohol, carbohydrate \rangle, \langle alligator, coconut \rangle,$
$\langle altar, sacristy \rangle, \langle object, library \rangle,$
$\langle object, pattern \rangle, \langle office, crew \rangle,$
$\langle onion, garlic \rangle, \langle saxophone, violin \rangle,$
$\langle saxophone, clarinet \rangle, \langle tongue, mouth \rangle,$
$\langle watercraft, boat \rangle, \langle watermelon, berry \rangle,$
$\langle weapon, warship \rangle, \langle wolf, coyote \rangle,$
$\langle wood, paper \rangle, \ldots$
$\}$

Количество извлеченных отношений

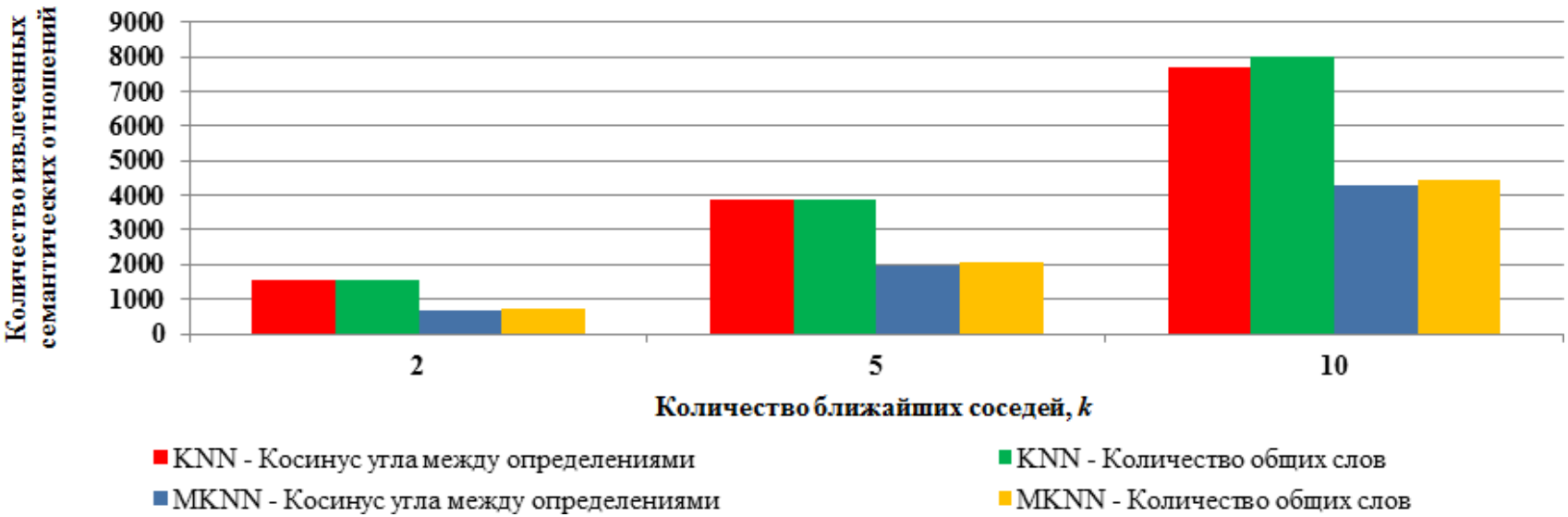


Рис.: Зависимость количество извлеченных отношений $|R|$ от количества ближайших соседей k.

Точность извлечения

Алгоритм	Мера подобия	Извлечено	Правильных	Точность
KNN	Cos	1548	1167	0.754
KNN	Overlap	1546	1176	0.761
MKNN	Cos	652	499	0.763
MKNN	Overlap	724	603	**0.833**

Таблица: Точность извлечения с помощью алгоритмов KNN и MKNN для k = 2 и 775 слов.

Альтернативные системы извлечения отношений

- SEXTANT (Grefensette, 1992) – точность извлечения 75%
- PMI-IR (Turney, 2001) – выбор 1 из 4 синонимов 74 %
- WikiRelate! (Strube and Ponzetto, 2006) – наиболее подобная система
  - не извлекает отношения
  - несколько другие метрики близости
  - использует решетку категорий Википедии
  - исходный код недоступен
  - корреляция около 0.59 с суждениями человека
- Explicit Semantic Analysis (Gabrilovich and Markovich, 2007)
- Wikipedia/Wiktionary (Zesch, Muller, and Gurevych, 2008)
- PF-IBF (Nakayama et al., 2007)

Заключение:

- Были предложены **новые методы** извлечения семантических отношений из Википедии с помощью алгоритмов ближайших соседей и двух метрик семантической близости.
- Наилучшие результаты (**83% точности**) показал метод MKNN с метрикой Overlap.
- Была реализована **система** с открытым исходным кодом Serelex, реализующая предложенные методы.
- Предложенные **методы характеризуются**:
  - вычислительной эффективностью
  - большим покрытием лексикона, за счет применения Википедии (3.8 млн. терминов на англ.)

Направления дальнейшего исследования:

- Применение разработанного метода для извлечения отношений на русском, французском и немецком языках.
- Повышение точности извлечения за счет анализа структуры полученного графа семантических отношений.