Моделирование и анализ информационных систем. Т. 23, № 2 (2016), с. 195–210
Modeling and Analysis of Information Systems. Vol. 23, No 2 (2016), pp. 195–210

DOI: 10.18255/1818-1015-2016-2-195-210

УДК 004.048

Коллективные потоковые вычисления: реляционные модели и алгоритмы

Усталов Д. А.

получена 2 апреля 2016

Аннотация. В последнее время краудсорсинг на основе выполения микрозадач получил широкое применение в области анализа неструктурированных данных. Разрабатываются специализированные методики, состоящие из множества этапов обработки исходных данных, требующих согласованности их представления для обеспечения воспроизводимости работы. Данная статья посвящена решению проблемы воспроизводимости и формализации процесса краудсорсинга микрозадачами. Предложена модель коллективных потоковых вычислений на основе расширенной реляционной модели и потоковой модели вычислений. Модель предназначена для обработки исходных данных в виде реляционных отношений путем параллельного выполнения этапов разметки микрозадачами и этапов автоматической синхронизации. Этапы обработки данных и связи между ними записываются с использованием схемы коллективных вычислений, представляющей собой слабо связный ориентированный ациклический граф. Описан синхронный алгоритм выполнения схем коллективных вычислений. Продемонстрированы приложения модели в области компьютерной лингвистики для уточнения лексикализации понятий в электронных тезаурусах и построения родо-видовых отношений между понятиями при помощи краудсорсинга. Процедура «добавить–удалить–подтвердить» позволяет внести в лексикализацию понятий недостающие лексемы и исключить посторонние. Процедура «род–вид–сопоставить» позволяет сформировать гипо-гиперонимические отношения между понятиями на основе соответствующих родо-видовых пар слов. Результаты экспериментов на материалах открытого электронного тезауруса русского языка подтверждают применимость разработанных процедур для развития лексических ресурсов. В экспериментах приняли участие как волонтеры из популярных социальных сетей, так и пользователи бирж краудсорсинга (за вознаграждение в форме микроплатежей).

Ключевые слова: краудсорсинг, потоковые вычисления, реляционная модель, компьютерная лингвистика

Для цитирования: Усталов Д. А., "Коллективные потоковые вычисления: реляционные модели и алгоритмы", *Моделирование и анализ информационных систем*, **23**:2 (2016), 195–210.

Об авторах:
Усталов Дмитрий Алексеевич, orcid.org/0000-0002-9979-2188, аспирант,
Институт математики и механики им. Н.Н. Красовского Уральского отделения Российской академии наук,
ул. Софьи Ковалевской, 16, г. Екатеринбург, 620990 Россия, e-mail: dau@imm.uran.ru

Благодарности:
Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 16-37-00354 мол_а «Методы автоматизации процесса коллективного построения лингвистических ресурсов». Исследование выполнено при финансовой поддержке РГНФ: проект «Новый открытый электронный тезаурус русского языка» № 13-04-12020 и проект «Интеграция тезаурусов RussNet и YARN» № 16-04-12019.

Введение

Краудсорсинг — способ получения услуг, идей и информации путём соучастия большого количества людей в Интернете [1], заслуживший большую популярность с момента своего появления в 2006 году. Зарубежные исследователи успешно применяют краудсорсинг для построения и разметки языковых ресурсов [2], в том числе лексических онтологий, словарей тональности и др. Отечественные исследователи применяют краудсорсинг преимущественно без денежного вознаграждения участников, вместо этого полагаясь на их *альтруизм* при построении открытых русскоязычных языковых ресурсов, таких как корпус текстов со снятой неоднозначностью [3], корпус текстов для перефразирования [4], электронный тезаурус [5]. В русскоязычной литературе множество участников процесса краудсорсинга получило название «толпа» (англ. *crowd*) [6].

Процесс коллективной работы осуществляется на специализированных платформах — биржах, где участники получают *микроплатежи* за работу — выполнение *микрозадач*. Оплата труда не гарантирует высокого качества результата [7], поэтому каждое задание выполняется несколькими разными участниками одновременно и затем проводится агрегация ответов на основе эвристического или вероятностного алгоритма. Известными биржами краудсорсинга являются MTurk[1], CrowdFlower[2] и «Яндекс.Толока[3]», с недавних пор доступная сторонним заказчикам.

Существует два основных способа повышения эффективности краудсорсинговой разметки: проектирование более совершенных подходов к представлению заданий и разработка математических методов обеспечения качества разметки. Объектом исследования данной работы является проблема представления данных и заданий для разметки перед участниками, не являющимися экспертами в заданной предметной области: биологии, географии, лингвистике и т. д.

В данной статье предложена модель коллективных потоковых вычислений, построенная на основе потоковых вычислений и расширенной реляционной модели. Модель предназначена для обработки исходных данных в виде реляционных отношений путём параллельного выполнения этапов коллективной разметки и этапов автоматической синхронизации. В разделе 1 представлен обзор литературы по тематике работы. В разделе 2 описана модель коллективных потоковых вычислений. В разделе 3 приведён алгоритм выполнения схем таких вычислений. В разделе 4 продемонстрированы приложения разработанных моделей для решения задач компьютерной лингвистики. В заключении подведены итоги работы и предложены пути дальнейших исследований.

1. Обзор литературы

Обзор литературы включает два направления: (1) модели коллективных вычислений и (2) методики решения задач при помощи краудсорсинга.

[1] https://www.mturk.com/mturk/welcome
[2] https://www.crowdflower.com/
[3] https://toloka.yandex.com/

1.1. Модели коллективных вычислений

Попытки адаптации популярных моделей компьютерного программирования к области краудсорсинга производятся с начала 2010-х гг., причём особое внимание уделяется различным вычислительным моделям распределённых и параллельных вычислений. Среди примеров следует отметить CrowdForge — адаптацию парадигмы MapReduce, в которой этапы *отображения* и *редукции* заменены на краудсорсинговые задачи [8], а также CrowdDB — расширение языка SQL, включающее оператор `CROWD` для запуска заданий на MTurk при выполнении запроса на манипулирование данными [9]. Также известны эксперименты по переносу различных нотаций моделирования бизнес-процессов и парадигм декларативного программирования [10].

Активное развитие технологий распределённых вычислений и снижение входных барьеров в данную область привело к росту популярности концепции потоковых вычислений [11], первоначально использованной при создании Манчестерской потоковой ЭВМ, а позднее — мультиклеточных процессоров [12]. Система CrowdWeaver адаптирует данную вычислительную модель для краудсорсинга [13], однако реализация данной системы, как и сведения о принципах её функционирования и используемой системе типов, закрыты.

Интерес к потоковой вычислительной модели проявляется у исследователей «Интернета вещей», использующих данную модель для интеграции датчиков различных производителей в одном помещении [14]. При описании потоковых вычислений применяют различные теоретико-графовые подходы, в том числе сети процессов Кана [15] и сети Петри [16].

1.2. Методики решения задач

Отдельного внимания заслуживают краудсорсинговые методики решения трудноформализуемых задач обработки и анализа неструктурированных данных.

Одной из самых известных методик краудсорсинга является Find-Fix-Verify, предложенная Бернштейном и соавторами для перефразирования текстовых документов [17]: на этапе *поиска* участники выделяют фрагменты исходного текста для последующего исправления, затем на этапе *исправления* участники предлагают варианты исправления для ранее выделенных фрагментов, после чего на этапе *проверки* участники голосуют за лучшие варианты из предложенных.

Норона и соавторы предложили методику PlateMate для определения калорийности пищи на основе фотографий [18]. Методика состоит из трёх последовательных этапов: на этапе *отметить* участники выделяют граничной рамкой блюда на фотографии, на этапе *опознать* участники выбирают из списка соответствующие рамкам блюда, на этапе *измерить* участники указывают размер порции каждого блюда.

Краудсорсинговая методика CrowdER предназначена для обнаружения дубликатов и связывания записей в базах данных различных товаров [19]. Задача решается в два этапа: сначала выполняется *оценка* подобия между парами товаров, затем осуществляется *проверка* пар записей, оставшихся после фильтрации.

При решении задач обработки естественного языка необходимы специализированные словари и иные языковые ресурсы, которые в последнее время всё чаще

создаются при помощи краудсорсинга [2]. Биманн предложил трёхэтапную методику Turk Bootstrap Word Sense Inventory (TWSI) для коллективного построения синсетов на основе явления лексической замещаемости слов [20]: на этапе *подстановки слов* участники предлагают синонимы для слов в заданном контексте, затем на этапе *выравнивания смыслов* участники оценивают семантическую близость пар слов, после чего на этапе *сопоставления значений* участники оценивают качество выведенных значений слов по примерам употребления.

2. Коллективные потоковые вычисления

На практике краудсорсинг на основе выполнения микрозадач предполагает обработку некоторого набора исходных данных в один или несколько этапов, причём множество таких этапов известно заранее. В рамках модели коллективных потоковых вычислений предлагается использовать реляционную модель данных, расширенную операциями вкладывания (англ. *nest*), выкладывания (англ. *unnest*) и расширенной проекции для упрощения работы с массивами и более сложными структурами данных [21,22]. Выбор реляционной модели обусловлен практическими соображениями: строгостью системы типов, популярностью модели, а также возможностью адаптации более новых документоориентированных подходов [23]. Введём понятие схемы коллективных вычислений для описания этапов обработки данных.

Определение 1. *Схема коллективных вычислений — слабо связный ориентированный ациклический граф W со множеством рёбер E, множество вершин которого образуется объединением множества этапов разметки S, множества этапов синхронизации Y и множества источников данных D.*

$$W = (S \cup Y \cup D, E) \tag{1}$$

Важно отметить, что все вершины W являются реляционными отношениями с заданными заголовками. Для удобства записи обозначим множество всех элементов схемы как $V = S \cup Y \cup D$ и множество входящих вершин в вершину $v \in V$ как $In(v)$. Обозначим заголовок отношения $v \in V$ как $H(v)$, тело как $B(v)$, а первичный ключ как $\mathrm{PK}(v) \subseteq H(v)$.

Определение 2. *Этап разметки $s \in S$ — это реляционное отношение, тело которого получено путём преобразования толпой участников кортежей единственного входящего отношения: $In(s) \subset V \wedge |In(s)| = 1$.*

Примером этапа разметки является задание на оценку семантической близости пары слов с одиночным выбором из вариантов «не связаны», «слабо связаны», «связаны» и «сильно связаны» [24]. Входным отношением в данном примере является таблица интересующих пар слов с заданным первичным ключом, а выходным отношением является таблица суждений участников о семантической близости каждой пары слов, образующая внутреннее соединение с исходной таблицей пар слов.

Определение 3. *Этап синхронизации $y \in Y$ — это реляционное отношение, тело которого получено путём автоматической обработки двух и более входящих отношений: $In(y) \subset V \wedge |In(y)| > 1$.*

Этапы синхронизации задают правила агрегации результатов выполнения этапов разметки с другими источниками данных, необходимыми для успешного решения исходной задачи. Примером этапа синхронизации является этап подготовки заданий по выравниванию смыслов методики TWSI, в которой для выбранных участниками лексических замещений подбираются примеры использования по корпусу текстов [20]. Входными отношениями являются таблица выявленных синонимов и табличное представление корпуса текстов. Выходным отношением является внутреннее соединение этих отношений с ограничением по количеству примеров на синоним.

Определение 4. *Источник данных $d \in D$ — это реляционное отношение, тело которого получено заранее и не зависит от других элементов схемы коллективных вычислений: $In(d) = \varnothing$.*

В качестве источников данных выступают словари, справочники, корпусы текстов, коллекции изображений и т. п. Источники данных получены заранее и не изменяются в процессе работы.

3. Выполнение схем коллективных вычислений

Этапы синхронизации и средства предварительной обработки данных реализуются в виде программного обеспечения различной сложности на разных языках программирования, что вносит дополнительные требования к детерминированности описания методики краудсорсинга. Введём понятие согласованности схемы коллективных вычислений.

Определение 5. *Согласованной схемой коллективных вычислений является схема, каждый кортеж каждого элемента которой однозначно идентифицирует породившие его кортежи.*

$$\forall v \in V \Big(\forall v' \in In(v) \big(H(v) \cap H(v') \supseteq \mathrm{PK}(v') \big) \Big). \tag{2}$$

В несогласованных схемах коллективных вычислений сведения между этапами либо передаются не целиком, что исключает возможность определить породившие кортежи, либо возникает неоднозначность из-за использования операций внешнего соединения без ограничений.

Алгоритм 1 осуществляет синхронное выполнение согласованных схем коллективных вычислений, обеспечивая удовлетворение зависимостей каждого элемента схемы. Каждый последующий элемент схемы не может быть выполнен до тех пор, пока каждая из его зависимостей не будет удовлетворена, т. е. завершена. Это обеспечивается применением битовой маски M, содержащей сведения о завершённости этапов: **false** — не завершён, **true** — завершён. При инициализации, битовая маска содержит значения **true** только для источников данных. В результате работы алгоритма такое значение получают все поля битовой маски.

Алгоритм 1 Синхронный алгоритм выполнения схем коллективных вычислений
Algorithm 1 Synchronous algorithm for executing collaborative computation workflows

Require: $V = S \cup Y \cup D, |D| > 0$
  for all $v \in V$ **do**
    $M_v \leftarrow (v \in D)$ ▷ Отметить все источники данных как завершённые.
  end for
  parallel for all $v \in V \setminus D$ **do**
    WAIT$(\forall v' \in In(v)(M_{v'} = \textbf{true}))$ ▷ Подождать завершение всех зависимостей.
    $B(v) \leftarrow$ RUN(v) ▷ Выполнить текущий элемент.
    $M_v \leftarrow \textbf{true}$ ▷ Отметить элемент как завершённый.
  end for
Ensure: $(\forall v \in V)(M_v = \textbf{true})$

Принцип работы алгоритма 1 аналогичен алгоритму обхода графа в ширину с несколькими отличиями: обход может начинаться с нескольких стартовых вершин, каждая вершина обрабатывается параллельно и только один раз, и каждая вершина ожидает пометки всех инцидентных ей вершин как завершённых. Требование $|D| > 0$ обеспечивает возможность обхода всех элементов схемы начиная с источников данных и наряду с условием ацикличности предотвращает взаимные блокировки при выполнении алгоритма.

Параллельная обработка достигается путём запуска каждой итерации рабочего цикла в отдельном потоке выполнения, что позволяет размещать задания соответствующих этапов разметки по мере выполнения предыдущих этапов. В данной работе подход к параллелизму реализован на основе механизма зелёных потоков [25], хотя возможно применение других решений: легковесных процессов, пулов потоков выполнения и т. п.

4. Приложения в компьютерной лингвистике

В данном разделе представлены две краудсорсинговые процедуры очистки и обогащения лексических ресурсов. Краудсорсинговая процедура «добавить–удалить–подтвердить» предназначена для уточнения лексикализации понятий в электронных тезаурусах и состоит из этапов *добавления* недостающих слов, *удаления* посторонних слов и *подтверждения* предложенных изменений [26]. Процедура «род–вид–сопоставить» предназначена для построения семантических отношений между понятиями: участниками соотносятся понятия в парах *род* и *вид*, после чего производится *сопоставление* подтверждённых пар [27]. Эксперименты проводились с использованием сервиса управления процессом краудсорсинга [28], развёрнутом в СКЦ ИММ УрО РАН[4]. Результаты обоих экспериментов доступны в виде открытых данных[5,6].

[4] http://parallel.uran.ru/
[5] http://ustalov.imm.uran.ru/pub/arc-yarn-100.tar.gz
[6] http://ustalov.imm.uran.ru/pub/gsm-ainl.tar.gz

4.1. Уточнение лексикализации понятий

Процедура «добавить–удалить–подтвердить» предназначена для повышения качества синонимических рядов в лексических ресурсах путём добавления в них недостающих слов и удаления из них лишних слов (рис. 1). Входными данными в данной процедуре является множество синсетов $\mathbb{S}$, подлежащих обработке, и множество слов-кандидатов $\mathbb{W}$ к включению в состав соответствующих синсетов. На каждом этапе решаются различные задачи:

- на этапе «добавить» (Add) участнику предоставляется синсет и список слов-кандидатов на включение в него. Участник выбирает недостающие слова для включения в синсет без искажения его общего значения;
- на этапе «удалить» (Remove) участнику предоставляется синсет. Участник может указать слова, искажающие общее значение синсета;
- на этапе «подтвердить» (Confirm) участнику предоставляется исходный синсет и его модифицированный вариант. Участник указывает лучший вариант из предложенных.

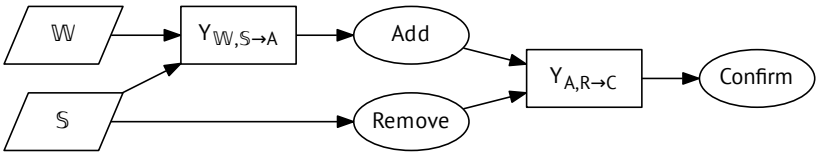

Рис. 1. Процедура «добавить–удалить–подтвердить»

Fig. 1. “Add–Remove–Confirm”

Результатом выполнения процедуры является множество модифицированных синсетов. Важно отметить, что этапы «добавить» и «удалить» выполняются одновременно и независимо друг от друга. Данная процедура записана в виде схемы коллективных вычислений ARC в формуле (3), описание отношений приведено в таблице 1. Подчёркивание при записи заголовка реляционного отношения обозначает элементы его первичного ключа. Поскольку атрибуты *words*, *added* и *removed* представляют собой списки слов, то при определении этапа синхронизации $Y_{A,R\to C}$ выполняются теоретико-множественные операции объединения и вычитания.

$$S_{ARC} = \{A, R, C\} \tag{3.1}$$

$$Y_{ARC} = \{Y_{\mathbb{W},\mathbb{S}\to A}, Y_{A,R\to C}\} \tag{3.2}$$

$$D_{ARC} = \{\mathbb{W}, \mathbb{S}\} \tag{3.3}$$

$$\begin{aligned} E_{ARC} = \big\{&(\mathbb{S}, Y_{\mathbb{W},\mathbb{S}\to A}), (\mathbb{W}, Y_{\mathbb{W},\mathbb{S}\to A}), (\mathbb{S}, R), (Y_{\mathbb{W},\mathbb{S}\to A}, A),\\ &(A, Y_{A,R\to C}), (R, Y_{A,R\to C}), (Y_{A,R\to C}, C)\big\} \end{aligned} \tag{3.4}$$

$$ARC = \big(S_{ARC} \cup Y_{ARC} \cup D_{ARC}, E_{ARC}\big) \tag{3.5}$$

Таблица 1. Элементы процедуры «добавить–удалить–подтвердить»
Table 1. Elements of "Add–Remove–Confirm"

Элемент	**Определение**
$\mathbb{W}$	$H(\mathbb{W}) = \{(\underline{sid}, \texttt{INT}), (candidates, \texttt{TEXT[]})\}$
$\mathbb{S}$	$H(\mathbb{S}) = \{(\underline{sid}, \texttt{INT}), (words, \texttt{TEXT[]})\}$
$Y_{\mathbb{W},\mathbb{S}\to A}$	$\mathbb{W} \bowtie \mathbb{S}$
A	$H(A) = \{(\underline{aid}, \texttt{INT}), (sid, \texttt{INT}), (added, \texttt{TEXT[]})\}$
R	$H(R) = \{(\underline{rid}, \texttt{INT}), (sid, \texttt{INT}), (removed, \texttt{TEXT[]})\}$
$Y_{A,R\to C}$	$\sigma_{words \neq words'}\big(\pi_{sid,aid,rid,words,\ \mathbb{S}.words \cup A.added \setminus R.removed \to words'}(\mathbb{S} \bowtie A \bowtie R)\big)$
C	$H(C) = \{(\underline{cid}, \texttt{INT}), (aid, \texttt{INT}), (rid, \texttt{INT}), (sid, \texttt{INT}), (b, \texttt{BOOL})\}$

Теорема 1. *Схема коллективных вычислений «добавить–удалить–подтвердить» является согласованной.*

Доказательство. Исходные данные представлены в отношениях $\mathbb{W}, \mathbb{S} \in D$. Проверим условие (2) для остальных отношений $V \setminus D$ по табл. 1.

1. $In(Y_{\mathbb{W},\mathbb{S}\to A}) = \{\mathbb{W}, \mathbb{S}\}$, причём $Y_{\mathbb{W},\mathbb{S}\to A}$ является проекцией $\mathbb{W} \bowtie \mathbb{S}$. Значит, $H(Y_{\mathbb{W},\mathbb{S}\to A}) \cap \big(H(\mathbb{W}) \cup H(\mathbb{S})\big) = \mathrm{PK}(\mathbb{W}) \cup \mathrm{PK}(\mathbb{S})$.

2. $In(A) = \{Y_{\mathbb{W},\mathbb{S}\to A}\}$, причём $H(A) \cap H(Y_{\mathbb{W},\mathbb{S}\to A}) = \mathrm{PK}(Y_{\mathbb{W},\mathbb{S}\to A}) = \mathrm{PK}(\mathbb{W}) \cup \mathrm{PK}(\mathbb{S})$.

3. $In(R) = \{\mathbb{S}\}$, причём $H(R) \cap H(\mathbb{S}) = \mathrm{PK}(\mathbb{S})$.

4. $In(Y_{A,R\to C}) = \{A, R\}$, причём $Y_{A,R\to C}$ является фильтрацией, заданной на проекции $\mathbb{S} \bowtie A \bowtie R$. Значит, $H(Y_{A,R\to C}) \cap \big(H(A) \cup H(R)\big) = \mathrm{PK}(A) \cup \mathrm{PK}(R)$.

5. $In(C) = \{Y_{A,R\to C}\}$, причём $H(C) \cap H(Y_{A,R\to C}) = \mathrm{PK}(Y_{A,R\to C}) = \mathrm{PK}(A) \cup \mathrm{PK}(R)$.

Каждый кортеж каждого отношения однозначно идентифицирует породившие его кортежи. Следовательно, схема (3) является согласованной. □

Исследование применимости процедуры «добавить–удалить–подтвердить» проводилось по материалам открытого электронного тезауруса русского языка Yet Another RussNet [5], поскольку он распространяется на условиях свободной лицензии Creative Commons (CC BY-SA), создан при помощи краудсорсинга, и содержит большое количество дублирующих друг друга понятий [5]. В качестве данных для эксперимента использовано подмножество тезауруса, состоящее из ста синсетов, для которых имеется большое количество дубликатов. Обнаружение дубликатов выполнялось эвристическим методом, объединяющим пару синсетов при наличии не менее двух общих слов [29].

Участники были приглашены из социальных сетей VK, Facebook и Twitter путём публикации открытого вызова, в результате чего время выполнения схемы коллективных вычислений (3) алгоритмом 1 составило 231 мин., в течение которых получено 1313 ответов на 300 заданий. В среднем на одно задание этапов «добавить» и «удалить» приходилось пять ответов, на одно задание «подтвердить» — три ответа. Ответы агрегировались простым голосованием большинства.

Для оценки качества результата были привлечены два эксперта. Каждый эксперт видел исходный синсет s и его модифицированную версию s'. От экспертов требовалось выставить оценку «0» в случае, если синсет s' не улучшился по сравнению с исходным синсетом s, или выставить оценку «1», если синсет существенно улучшился в результате применения процедуры. Синсеты, которые, по мнению участников, не изменились в ходе разметки, были исключены из процесса оценки. Таким образом, осталось всего 84 синсета из 100.

Оба эксперта установили, что $\frac{70}{84} = 83\,\%$ оставшихся синсетов существенно улучшились по результатам эксперимента (Таблица 2), хотя согласованность их суждений по значению коэффициента каппа Флейса [30] достаточно низка ($\kappa = 0,143$). Это обусловлено скошенностью распределения ответов: 14 оценок «0» и 70 за оценку «1», что является известной проблемой данного коэффициента [31].

Таблица 2. Результаты эксперимента «добавить–удалить–подтвердить»
Table 2. Results for “Add–Remove–Confirm”

Участники		**Эксперты**	
Изменилось	84	Улучшилось	70
Не изменилось	16	Не улучшилось	14
Всего	100	**Всего**	84

Несмотря на это, ответы экспертов согласуются хорошо: различается лишь 20 суждений из 84, что подтверждается индексом Жаккара, равным $1 - \frac{20}{84} = 74\,\%$. Только $\frac{4}{84} = 5\,\%$ всех синсетов отмечены обоими экспертами как «не улучшившиеся». Каждая из этих ошибок принадлежит разному классу:

- участник перепутал гипероним и синоним;
- участник перепутал гипоним и синоним;
- участник перепутал когипоним и синоним;
- добавлено слово, явно не соответствующее синсету.

Это показывает, что участники, не являющиеся специалистами-лексикографами, не всегда способны корректно разделить отношение синонимии от отношения гипо- и гиперонимии. На практике слова часто замещаются гиперонимами, например: «*Лев* был голоден. *Зверь* рычал.». Оценки экспертов различались в $\frac{20}{84} = 24\,\%$ случаев, разделяемых на три класса:

- синсет, описывающий два или более понятий, остался неоднозначным даже после удаления лишних слов;

- большое количество корректных синонимов добавлено к синсету, однако лишнее слово не было удалено;
- добавленные слова образуют достаточно хороший синсет, однако его значение отличается от значения исходного синсета.

Полученные результаты подтверждают применимость процедуры «добавить–удалить–подтвердить» для уточнения лексикализации понятий в электронных тезаурусах.

4.2. Построение родо-видовых отношений

Процедура «род–вид–сопоставить» предназначена для построения родо-видовых отношений между понятиями в электронных тезаурусах (рис. 2). Входными данными в данной процедуре является множество синсетов $\mathbb{S}$ и множество родо-видовых пар между словами $\mathbb{R}$. Этапы процедуры предполагают решение следующих задач:

- на этапе «род» (Genus) задана родо-видовая пара слов и синсет. Участник должен подтвердить, что синсет корректно представляет *род* для слова из пары;
- на этапе «вид» (Species) задана родо-видовая пара слов и синсет. Участник должен подтвердить, что синсет корректно представляет *вид* для слова из пары;
- на этапе «сопоставить» (Match) задана пара синсетов. Участник должен подтвердить, что данная пара синсетов образует осмысленное родо-видовое отношение.

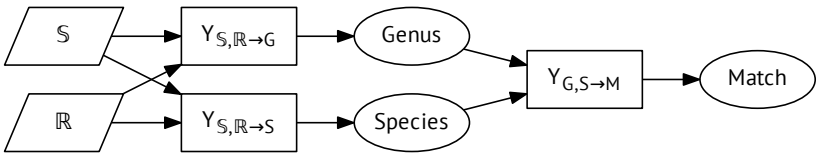

Рис. 2. Процедура «род–вид–сопоставить»

Fig. 2. “Genus–Species–Match”

Результатом выполнения процедуры является множество родо-видовых пар между синсетами. Этапы «род» и «вид» выполняются одновременно и независимо друг от друга. Данная процедура записана в виде схемы коллективных вычислений *GSM* в формуле (4), описания отношений приведены в таблице 3. Запись $\mu(\cdot)$ обозначает операцию выкладывания, упомянутую в разделе 2.

$$S_{GSM} = \{G, S, M\} \tag{4.1}$$

$$Y_{GSM} = \{Y_{\mathbb{S},\mathbb{R}\to G}, Y_{\mathbb{S},\mathbb{R}\to S}, Y_{G,S\to M}\} \tag{4.2}$$

$$D_{GSM} = \{\mathbb{S}, \mathbb{R}\} \tag{4.3}$$

$$E_{GSM} = \big\{(\mathbb{S}, Y_{\mathbb{S},\mathbb{R}\to G}), (\mathbb{S}, Y_{\mathbb{S},\mathbb{R}\to S}), (\mathbb{R}, Y_{\mathbb{S},\mathbb{R}\to G}), (\mathbb{R}, Y_{\mathbb{S},\mathbb{R}\to S}), (Y_{\mathbb{S},\mathbb{R}\to G}, G), (Y_{\mathbb{S},\mathbb{R}\to S}, S), (S, Y_{G,S\to M}), (G, Y_{G,S\to M}), (Y_{G,S\to M}, M)\big\} \tag{4.4}$$

$$GSM = \big(S_{GSM} \cup Y_{GSM} \cup D_{GSM}, E_{GSM}\big) \tag{4.5}$$

Таблица 3. Элементы процедуры «род–вид–сопоставить»
Table 3. Elements of “Genus–Species–Match”

Элемент	**Определение**
$\mathbb{S}$	$H(\mathbb{S}) = \big\{(\underline{sid}, \texttt{INT}), (words, \texttt{TEXT[]})\big\}$
$\mathbb{R}$	$H(\mathbb{R}) = \big\{(\underline{hypernym}, \texttt{TEXT}), (\underline{hyponym}, \texttt{TEXT})\big\}$
$Y_{\mathbb{S},\mathbb{R}\to G}$	$\mathbb{R} \bowtie \pi_{sid,words,\mu(words)\to hypernym}(\mathbb{S})$
$Y_{\mathbb{S},\mathbb{R}\to S}$	$\mathbb{R} \bowtie \pi_{sid,words,\mu(words)\to hyponym}(\mathbb{S})$
G	$H(G) = \big\{(\underline{gid}, \texttt{INT}), (sid, \texttt{INT}), (hypernym, \texttt{TEXT}), (hyponym, \texttt{TEXT}), (b, \texttt{BOOL})\big\}$
S	$H(S) = \big\{(\underline{spid}, \texttt{INT}), (sid, \texttt{INT}), (hypernym, \texttt{TEXT}), (hyponym, \texttt{TEXT}), (b, \texttt{BOOL})\big\}$
$Y_{G,S\to M}$	$\sigma_{\substack{sid_1\neq sid_2 \\ \wedge b_1=b_2=\textbf{true}}} \big(\pi_{\substack{gid,sid\to sid_1,b\to b_1, \\ hypernym,hyponym}}(G) \underset{\substack{G.hypernym=S.hypernym \\ \wedge G.hyponym=S.hyponym}}{\bowtie} \pi_{\substack{spid,sid\to sid_2,b\to b_2, \\ hypernym,hyponym}}(S)\big)$
M	$H(M) = \big\{(\underline{mid}, \texttt{INT}), (gid, \texttt{INT}), (spid, \texttt{INT}), (b, \texttt{BOOL})\big\}$

Теорема 2. *Схема коллективных вычислений «род–вид–сопоставить» является согласованной.*

Доказательство. Исходные данные представлены в отношениях $\mathbb{S}, \mathbb{R} \in D$. Как и при доказательстве теоремы 1, проверим условие (2) для остальных отношений $V \setminus D$ по табл. 3.

1. $In(Y_{\mathbb{S},\mathbb{R}\to G}) = \{\mathbb{S}, \mathbb{R}\}$, причём $Y_{\mathbb{S},\mathbb{R}\to G}$ является естественным соединением $\mathbb{S}$ и $\mathbb{R}$. Значит, $H(Y_{\mathbb{S},\mathbb{R}\to G}) \cap \big(H(\mathbb{S}) \cup H(\mathbb{R})\big) = \mathrm{PK}(\mathbb{S}) \cup \mathrm{PK}(\mathbb{R})$. Аналогичное утверждение справедливо для $Y_{\mathbb{S},\mathbb{R}\to S}$ с точностью до обозначений.

2. $In(G) = \{Y_{\mathbb{S},\mathbb{R}\to G}\}$, причём $H(G) \cap H(Y_{\mathbb{S},\mathbb{R}\to G}) = \mathrm{PK}(Y_{\mathbb{S},\mathbb{R}\to G}) = \mathrm{PK}(\mathbb{S}) \cup \mathrm{PK}(\mathbb{R})$. Аналогичное утверждение справедливо для S с точностью до обозначений.

3. $In(Y_{G,S\to M}) = \{G, S\}$, причём $Y_{G,S\to M}$ является фильтрацией, заданной на проекции тета-соединения G и S по $\mathrm{PK}(\mathbb{R})$. Следовательно, $H(Y_{G,S\to M}) \cap \big(H(G) \cup H(S)\big) = \mathrm{PK}(G) \cup \mathrm{PK}(S)$ с учётом переименований.

4. $In(M) = \{Y_{G,S\to M}\}$, причём $H(M) \cap H(Y_{G,S\to M}) = \mathrm{PK}(Y_{G,S\to M}) = \mathrm{PK}(G) \cup \mathrm{PK}(S)$.

Каждый кортеж каждого отношения однозначно идентифицирует породившие его кортежи. Следовательно, схема (4) является согласованной. □

Исследование применимости процедуры «род–вид–сопоставить» проводилось по материалам открытого электронного тезауруса русского языка Yet Another RussNet [5]. При подготовке данных осуществлялось сопоставление синсетов со словарём предметной области «безопасность жизнедеятельности[7]»: набор данных для построения родо-видовых отношений состоял из 2271 синсета и 383 кандидатов-отношений.

В отличие от предыдущего эксперимента с привлечением волонтёров из социальных сетей, данный эксперимент проводился на краудсорсинговой бирже TurboText[8], поддерживающей размещение и выполнение микрозадач. Минимальное вознаграждение участника на данной бирже составляет 4 руб., среднее вознаграждение на момент первого октября 2015 г. — 11 руб. Для удобства разметки, задания группировались в пакеты из нескольких заданий. Вознаграждение участников на трёх этапах выполнения данной процедуры составляло 5 руб. Каждое задание выполнялось не менее чем пятью различными участниками. При отображении многословных синсетов участникам демонстрировались только первые пять слов, отсортированные по убыванию частоты появления в Национальном корпусе русского языка [32].

Выполнение схемы коллективных вычислений (4) алгоритмом 1 заняло 264 мин. Получено 13 056 ответов на 2558 заданий; на каждое задание собиралось пять ответов от разных участников. На этапах «род» и «вид» ответы агрегировались простым голосованием большинства; на финальном этапе «сопоставить» было проведено сравнение нескольких методов статистического вывода ответов: MV — голос большинства, KOS — итеративный алгоритм Каргера–Оха–Шаха, ZenCrowd — вероятностный метод на основе фактор-графов [27, с. 122].

Отдельный интерес представлял вопрос целесообразности применения краудсорсинга для решения такой задачи, поскольку сведения о синонимических рядах и родо-видовых парах слов могут быть использованы для автоматического построения отношений эвристическим образом. В заданной родо-видовой паре $(g, s) \in \mathbb{R}$, представленной множеством родовых синсетов $\mathbb{S}_g \subseteq \mathbb{S}$ и видовых синсетов $\mathbb{S}_s \subseteq \mathbb{S}$, между парой синсетов $S_g \in \mathbb{S}_g$ и $S_s \in \mathbb{S}_s$ строится отношение, если существует хотя бы ещё одна родо-видовая пара, связывающая эти синсеты:

$$\left|\{s' : \exists(g, s') \in \mathbb{R}\} \cap \{g' : \exists(g', s) \in \mathbb{R}\}\right| > 1. \tag{5}$$

Полученные результаты были обработаны экспертом: определено количество верных положительных TP, верных отрицательных TN, ложных положительных FP и ложных отрицательных FN ответов, по которым вычислена точность P, полнота R и F_1-мера (табл. 4). Дополнительно результаты выполнения эвристического метода (5) приведены в строке «Эвристика».

Исследование результатов выявило три типа ошибок:

- ошибки участников, состоящие в некорректном понимании лексических значений в заданиях;
- ошибки в синсетах, вызванные чрезмерной общностью или узостью выраженных в них понятий;

[7] http://www.mchs.gov.ru/dop/terms
[8] http://www.turbotext.ru/

Таблица 4. Результаты эксперимента «род–вид–сопоставить»
Table 4. Results for “Genus–Species–Match”

Метод	**TP**	**TN**	**FP**	**FN**	**P**	**R**	**F_1**
Эвристика	40	102	17	128	0,70	0,24	0,36
MV	129	57	62	39	0,68	0,77	0,72
KOS	142	63	56	26	0,72	0,84	0,78
ZenCrowd	146	69	50	22	**0,74**	**0,87**	**0,80**

- ошибки в данных, обусловленные некорректными исходными данными из недоверенных источников, например слово «город» в качестве гиперонима в родовидовой паре (место, город), или наличием таких бессмысленных синсетов, как {страна, столица, область, район}.

Несмотря на то, что каждое задание было выполнено не менее чем пятью разными участниками, доля совпавших ответов низка и составляет 55 %. При этом процедура проявила себя устойчивой к ошибкам, хотя их количество можно снизить путём дополнительной очистки исходных данных. Полученные результаты подтверждают применимость процедуры «род–вид–сопоставить» для построения родо-видовых отношений между синсетами.

Заключение

Предложенная модель коллективных потоковых вычислений основана на потоковой вычислительной модели и расширенной реляционной модели для представления и выполнения схем решения различных задач анализа и обработки данных при помощи краудсорсинга. На основе данной модели разработаны краудсорсинговые процедуры «добавить–удалить–подтвердить» — для уточнения лексикализации понятий и «род–вид–сопоставить» — для построения родо-видовых отношений между ними. Результаты проведённых экспериментов подтверждают применимость этих процедур для развития электронных лексических ресурсов. В качестве следующего шага данного исследования будет проведено построение открытого тезауруса предметной области при помощи предложенных процедур.

Большой интерес для дальнейшей работы представляют два направления исследований: интеграция с медицинскими технологиями и адаптация концепций из области распределённых вычислений. Применение электроэнцефалографии и иных электрофизиологических методов исследования функционального состояния головного мозга открывает новые перспективы как для изучения сложности заданий для участников, так и для разработки более совершенных человеко-машинных интерфейсов [33]. Развитие механистических подходов к краудсорсингу позволит создать ещё более эффективные процедуры решения важных задач анализа, обработки и представления данных в самых разных областях науки и промышленности [34].

Благодарности. При выполнении экспериментов использована инфраструктура суперкомпьютера «Уран» ИММ УрО РАН. Автор благодарит А. В. Созыкина и

Д. И. Игнатова за ценные замечания по содержанию работы, а также Ю. А. Киселёва за совместное проведение эксперимента «добавить–удалить–подтвердить» в работе [26].

Список литературы / References

[1] Estellés-Arolas E., González-Ladrón-de Guevara F., "Towards an integrated crowdsourcing definition", *Journal of Information Science*, **38**:2 (2012), 189–200, http://jis.sagepub.com/content/38/2/189.

[2] *The People's Web Meets NLP*, eds. Gurevych I., Kim J., Springer Berlin Heidelberg, 2013, http://dx.doi.org/10.1007/978-3-642-35085-6.

[3] Bocharov V., Alexeeva S., Granovsky D. et al., "Crowdsourcing morphological annotation", Computational Linguistics and Intellectual Technologies: papers from the Annual conference "Dialogue", 2013, 109–124, http://www.dialog-21.ru/digests/dialog2013/materials/pdf/BocharovVV.pdf.

[4] Pronoza E., Yagunova E., "Comparison of Sentence Similarity Measures for Russian Paraphrase Identification", Proceedings of the AINL-ISMW FRUCT, 2015, 74–82, http://dx.doi.org/10.1109/AINL-ISMW-FRUCT.2015.7382973.

[5] Braslavski P., Ustalov D., Mukhin M., "A Spinning Wheel for YARN: User Interface for a Crowdsourced Thesaurus", Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics, 2014, 101–104, http://www.aclweb.org/anthology/E/E14/E14-2026.pdf.

[6] Шуровьески Дж., *Мудрость толпы*, Манн, Иванов и Фербер, 2013, http://www.mann-ivanov-ferber.ru/books/paperbook/the-wisdom-of-crowds/; [Surowiecki J., *The Wisdom of Crowds*, Doubleday, 2004, (in Russian).]

[7] Gadiraju U., Kawase R., Dietze S. et al., "Understanding Malicious Behavior in Crowdsourcing Platforms: The Case of Online Surveys", Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, 2015, 1631–1640, http://dx.doi.org/10.1145/2702123.2702443.

[8] Kittur A., Smus B., Khamkar S., et al., "CrowdForge: Crowdsourcing Complex Work", Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 2011, 43–52, http://dx.doi.org/10.1145/2047196.2047202.

[9] Franklin M. J., Kossmann D., Kraska T. et al., "CrowdDB: Answering Queries with Crowdsourcing", Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, 2011, 61–72, http://dx.doi.org/10.1145/1989323.1989331.

[10] Kucherbaev P., Daniel F., Tranquillini S. et al., "Crowdsourcing Processes: A Survey of Approaches and Opportunities", *IEEE Internet Computing*, **20**:2 (2016), 50–56, http://dx.doi.org/10.1109/MIC.2015.96.

[11] Johnston W. M., Hanna J. R. P., Millar R. J., "Advances in Dataflow Programming Languages", *ACM Comput. Surv.*, **36**:1 (2004), 1–34, http://dx.doi.org/10.1145/1013208.1013209.

[12] Стрельцов Н. В., "Архитектура и реализация мультиклеточных процессоров", Труды V Международной научной конференции «Параллельные вычисления и задачи управления» (Москва, 26–28 октября 2010 г.), 2010, 1087–1104; [Streltsov N. V., "Arkhitektura i realizatsiya multikletochnykh protsessorov", Trudy V Mezhdunarodnoy nauchnoy konferentsii "Parallelnye vychisleniya i zadachi upravleniya" (Moskva, 26–28 oktyabrya 2010 g.), 2010, 1087–1104, (in Russian).]

[13] Kittur A., Khamkar S., André P., "CrowdWeaver: Visually Managing Complex Crowd Work", Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, 2012, 1033–1036, http://dx.doi.org/10.1145/2145204.2145357.

[14] Giang N. K., Blackstock M., Lea R. et al., “Distributed Data Flow: A Programming Model for the Crowdsourced Internet of Things”, Proceedings of the Doctoral Symposium of the 16th International Middleware Conference, 4:1–4:4, http://dx.doi.org/10.1145/2843966.2843970.

[15] Kahn G., “The semantics of a simple language for parallel programming”, *Information Processing*, **74** (1974), 471–475.

[16] Murata T., “Petri Nets: Properties, Analysis and Applications”, *Proceedings of the IEEE*, **77**:4 (1989), 541–580, http://dx.doi.org/10.1109/5.24143.

[17] Bernstein M. S., Little G., Miller R. C. et al., “Soylent: A Word Processor with a Crowd Inside”, Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology, 2010, 313–322, http://dx.doi.org/10.1145/1866029.1866078.

[18] Noronha J., Hysen E., Zhang H. et al., “PlateMate: Crowdsourcing Nutritional Analysis from Food Photographs”, Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 2011, 1–12, http://dx.doi.org/10.1145/2047196.2047198.

[19] Wang J., Kraska T., Franklin M. J. et al., “CrowdER: Crowdsourcing Entity Resolution”, Proc. VLDB Endow., **5**:11 (2012), 1483–1494, http://dx.doi.org/10.14778/2350229.2350263.

[20] Biemann C., “Creating a system for lexical substitutions from scratch using crowdsourcing”, *Language Resources and Evaluation*, **47**:1 (2013), 97–122, http://dx.doi.org/10.1007/s10579-012-9180-5.

[21] Schek H.-J., Scholl M. H., “The relational model with relation-valued attributes”, *Information Systems*, **11**:2 (1986), 137–147, http://dx.doi.org/10.1016/0306-4379(86)90003-7.

[22] Garcia-Molina H., Ullman J. D., Widom J., *Database Systems: The Complete Book*, 2nd edition, Prentice Hall Press, 2008.

[23] Zhao G., Huang W., Liang S. et al., “Modeling MongoDB with Relational Model”, Emerging Intelligent Data and Web Technologies (EIDWT), 2013 Fourth International Conference on, 2013, 115–121, http://dx.doi.org/10.1109/EIDWT.2013.25.

[24] Panchenko A., Loukachevitch N. V., Ustalov D. et al., “RUSSE: The First Workshop on Russian Semantic Similarity”, Computational Linguistics and Intellectual Technologies: papers from the Annual conference “Dialogue”, **2** (2015), 89–105, http://www.dialog-21.ru/digests/dialog2015/materials/pdf/PanchenkoAetal.pdf.

[25] McCartney W. P., Sridhar N., “Abstractions for Safe Concurrent Programming in Networked Embedded Systems”, Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, 2006, 167–180, http://dx.doi.org/10.1145/1182807.1182825.

[26] Ustalov D., Kiselev Y., “Add-Remove-Confirm: Crowdsourcing Synset Cleansing”, Application of Information and Communication Technologies (AICT), 2015 IEEE 9th International Conference on, 2015, 143–147, http://dx.doi.org/10.1109/ICAICT.2015.7338534.

[27] Ustalov D., “Crowdsourcing Synset Relations with Genus-Species-Match”, Proceedings of the AINL-ISMW FRUCT, 2015, 118–124, http://dx.doi.org/10.1109/AINL-ISMW-FRUCT.2015.7382980.

[28] Ustalov D., “A Crowdsourcing Engine for Mechanized Labor”, *Proceedings of the Institute for System Programming*, **27**:3 (2015), 351–364, http://dx.doi.org/10.15514/ISPRAS-2015-27(3)-25.

[29] Kiselev Y., Ustalov D., Porshnev S., “Eliminating Fuzzy Duplicates in Crowdsourced Lexical Resources”, Proceedings of the Eighth Global Wordnet Conference, 2016, 161–167, http://gwc2016.racai.ro/procedings.pdf.

[30] Fleiss J. L., Levin B., Paik M. C., *Statistical Methods for Rates and Proportions*, 3rd edition, John Wiley & Sons, 2003.

[31] Powers D. M. W., "The Problem with Kappa", Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, 2012, 345–355, http://aclweb.org/anthology/E12-1035.

[32] Ляшевская О. Н., Шаров С. А., *Частотный словарь современного русского языка (на материалах Национального корпуса русского языка)*, Азбуковник, 2009; [Lyashevskaya O. N., Sharoff S. A., *Chastotnyy slovar sovremennogo russkogo yazyka (na materialakh Natsionalnogo korpusa russkogo yazyka)*, Azbukovnik, 2009, (in Russian).]

[33] Healy G. F., Gurrin C., Smeaton A. F., "Informed Perspectives on Human Annotation Using Neural Signals", *MultiMedia Modeling: 22nd International Conference, Proceedings, Part II*, Lecture Notes in Computer Science, **9517**, 2016, 315–327, http://dx.doi.org/10.1007/978-3-319-27674-8_28.

[34] Ignatov D. I., Kaminskaya A. Yu., Bezzubtseva A. A. et al., "FCA-Based Models and a Prototype Data Analysis System for Crowdsourcing Platforms", *Conceptual Structures for STEM Research and Education*, Lecture Notes in Computer Science, **7735**, 2013, 173–192, http://dx.doi.org/10.1007/978-3-642-35786-2_13.

Ustalov D. A., "Dataflow-Driven Crowdsourcing: Relational Models and Algorithms", *Modeling and Analysis of Information Systems*, **23**:2 (2016), 195–210.

DOI: 10.18255/1818-1015-2016-2-195-210

Abstract. Recently, microtask crowdsourcing has become a popular approach for addressing various data mining problems. Crowdsourcing workflows for approaching such problems are composed of several data processing stages which require consistent representation for making the work reproducible. This paper is devoted to the problem of reproducibility and formalization of the microtask crowdsourcing process. A computational model for microtask crowdsourcing based on an extended relational model and a dataflow computational model has been proposed. The proposed collaborative dataflow computational model is designed for processing the input data sources by executing annotation stages and automatic synchronization stages simultaneously. Data processing stages and connections between them are expressed by using collaborative computation workflows represented as loosely connected directed acyclic graphs. A synchronous algorithm for executing such workflows has been described. The computational model has been evaluated by applying it to two tasks from the computational linguistics field: concept lexicalization refining in electronic thesauri and establishing hierarchical relations between such concepts. The "Add–Remove–Confirm" procedure is designed for adding the missing lexemes to the concepts while removing the odd ones. The "Genus–Species–Match" procedure is designed for establishing "is-a" relations between the concepts provided with the corresponding word pairs. The experiments involving both volunteers from popular online social networks and paid workers from crowdsourcing marketplaces confirm applicability of these procedures for enhancing lexical resources.

Keywords: crowdsourcing, dataflow model, relational model, computational linguistics

On the authors:
Ustalov Dmitry Alekseevich, orcid.org/0000-0002-9979-2188, graduate student,
N.N. Krasovskii Institute of Mathematics and Mechanics of the Ural Branch of the Russian Academy of Sciences, Sofia Kovalevskaya str., 16, Yekaterinburg, 620990, Russia,
e-mail: dau@imm.uran.ru

Acknowledgments:
The reported study was funded by RFBR according to the research project No. 16-37-00354 мол_а "Adaptive Crowdsourcing Methods for Linguistic Resources". This work was supported by the Russian Foundation for the Humanities project no. 13-04-12020 "New Open Electronic Thesaurus for Russian" and project no. 16-04-12019 "RussNet and YARN thesauri integration".