

Анализ данных пользователей социальных сетей

Антон Коршунов

Институт системного программирования
Российской Академии Наук

30 мая 2013 года

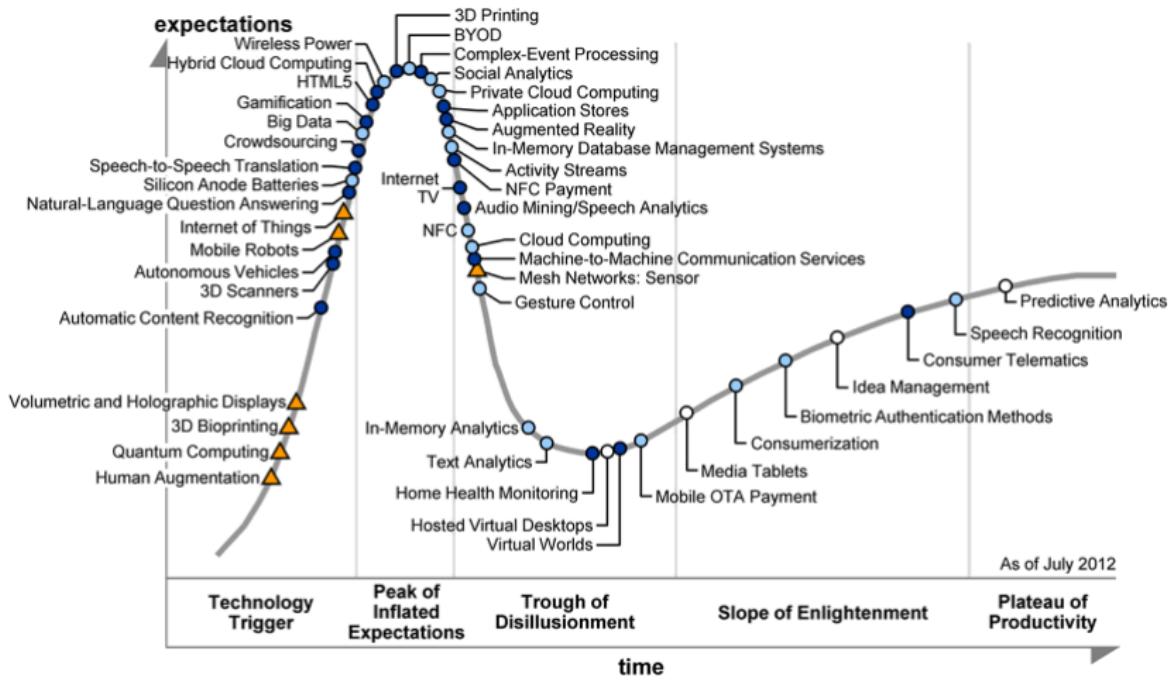
Анализ социальных сетей

Актуальность

- ▶ Уникальный источник данных о личной жизни и интересах реальных людей: переписка, дневники, фотоальбомы
- ▶ Проникновение во все сферы деятельности: нишевые сети IT-специалистов, трейдеров, бухгалтеров
- ▶ Социализация контента: фото, видео, музыка, новости, рецепты
- ▶ Социализация сервисов: магазины, форумы, рекомендации
- ▶ Растущий спрос на социальные сервисы, упрощающие общение и обмен информацией
- ▶ Беспрецедентные возможности для таргетированного маркетинга

Анализ социальных сетей

Gartner Hype Cycle for Emerging Technologies 2012



Plateau will be reached in:

O less than 2 years

● 2 to 5 years

● 5 to 10 years

▲ more than 10 years

✖ obsolete
✖ before plateau

Анализ социальных сетей

Объединение множества дисциплин

- ▶ Теория графов
- ▶ Анализ текстов на естественном языке
- ▶ Информационный поиск
- ▶ Машинное обучение и искусственный интеллект
- ▶ Распределённые алгоритмы и системы
- ▶ Теория игр
- ▶ Социология и антропология
- ▶ Психология
- ▶ Маркетинг
- ▶ ...

Анализ социальных сетей

Проблемы

- ▶ Обеспечение приватности персональных данных
- ▶ Нестабильность качества пользовательского контента (спам и ложные аккаунты)
- ▶ Частые обновления пользовательской модели и функционала
- ▶ Оптимизация распределённого хранения и обработки
- ▶ Появление новых бизнес-задач, требующих срочного решения

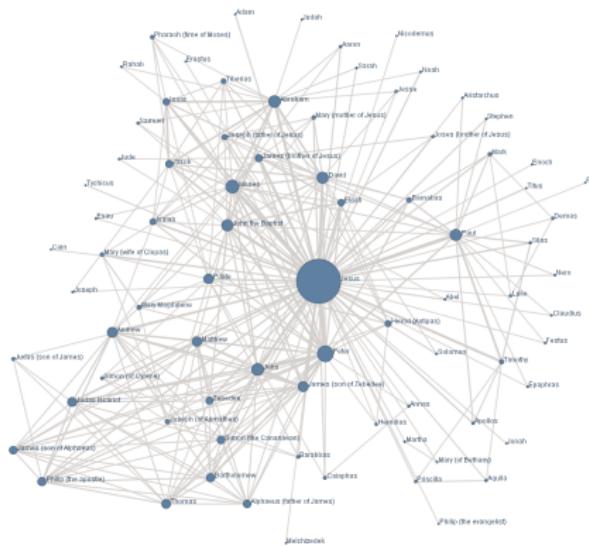
Анализ социальных сетей

Потребности

- ▶ Изучение закономерностей общения и обмена информацией
- ▶ Совместная обработка разнородных данных: граф, тексты сообщений, фото, видео, музыка, логи
- ▶ Новые математические модели для тестирования гипотез
- ▶ Эффективные алгоритмы для решения новых задач
- ▶ Новые сервисы для облегчения коммуникации

Социальный граф

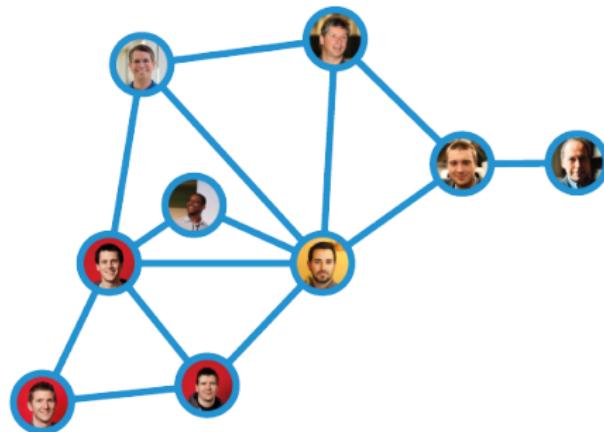
Оффлайн-эра



- ▶ **Метод:** интервью
- ▶ **Формат:** связи и несколько атрибутов (имя, возраст, профессия)
- ▶ **Результат:** небольшие разрозненные графы

Социальный граф

1990-2000-е



Базовая персонализация и функциональность

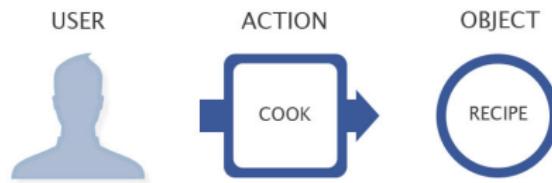
- ▶ пользовательские профили
- ▶ ненаправленные связи с семантикой дружбы
- ▶ личные сообщения
- ▶ обмен информацией

Социальный граф

Современное состояние

Новые типы узлов

- ▶ контент: рисунки, видео, музыка, геолокации, хэштеги, внешние URL
- ▶ сообщества, определяемые пользователями: списки, группы



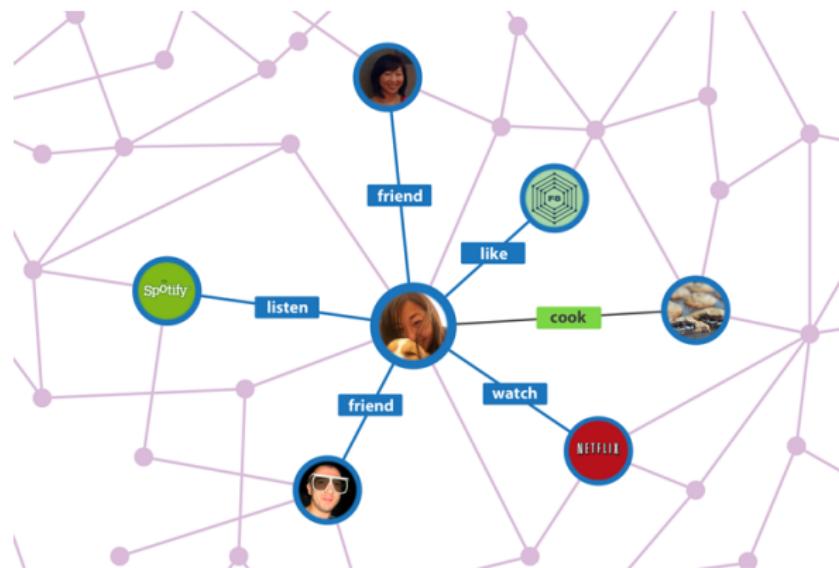
Новые типы рёбер

- ▶ направленные парасоциальные связи: фолловеры, фанаты, подписчики
- ▶ ориентация на контент: репосты, комментарии, избранное, "мне нравится"
- ▶ логи активности: посещения страницы, просмотры видео, прослушивания музыкального трека

Социальный граф

Современное состояние

Facebook OpenGraph



Всё связано со всем

Классификация

Сети общего назначения

- ▶ Поддержание оффлайновых связей
- ▶ Новые знакомства
- ▶ Обсуждение повседневных событий
- ▶ Развлечения



одноклассники

Классификация

Нишевые сети

- ▶ Поддержание профессиональных контактов
- ▶ Новые контакты с единомышленниками
- ▶ Обсуждение общих интересов



comon
Соцсеть трейдеров

Классификация

Контентные сети

- ▶ Обмен контентом
- ▶ Распространение новостей
- ▶ Создание и развитие сообществ по интересам
- ▶ Развлечения



Классификация

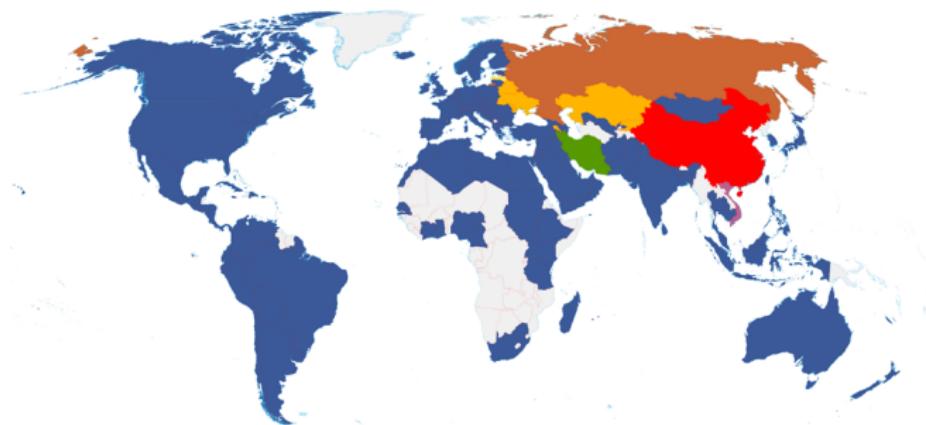
Другие

- ▶ Геосоциальные
- ▶ Закладки, каталоги
- ▶ Игровые



Популярность

WORLD MAP OF SOCIAL NETWORKS June 2012



■ Facebook ■ QZone ■ VKontakte ■ Odnoklassniki ■ Draugiem
■ Zing ■ Cloob

credits: Vincenzo Cosenza vincos.it

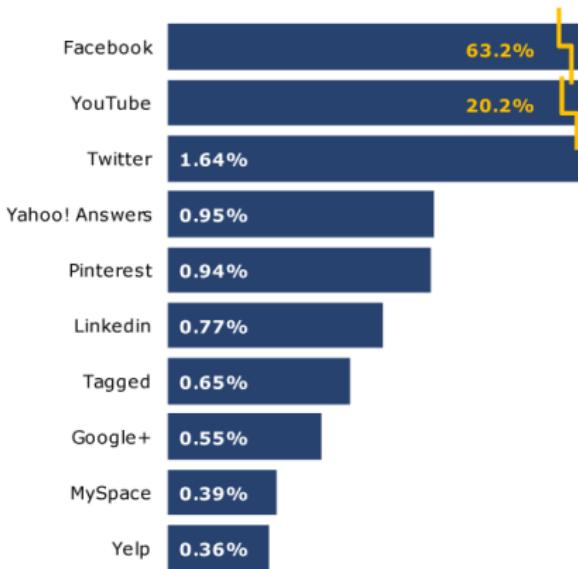
license: CC-BY-NC

sources: Google Trends for Websites/Alexa

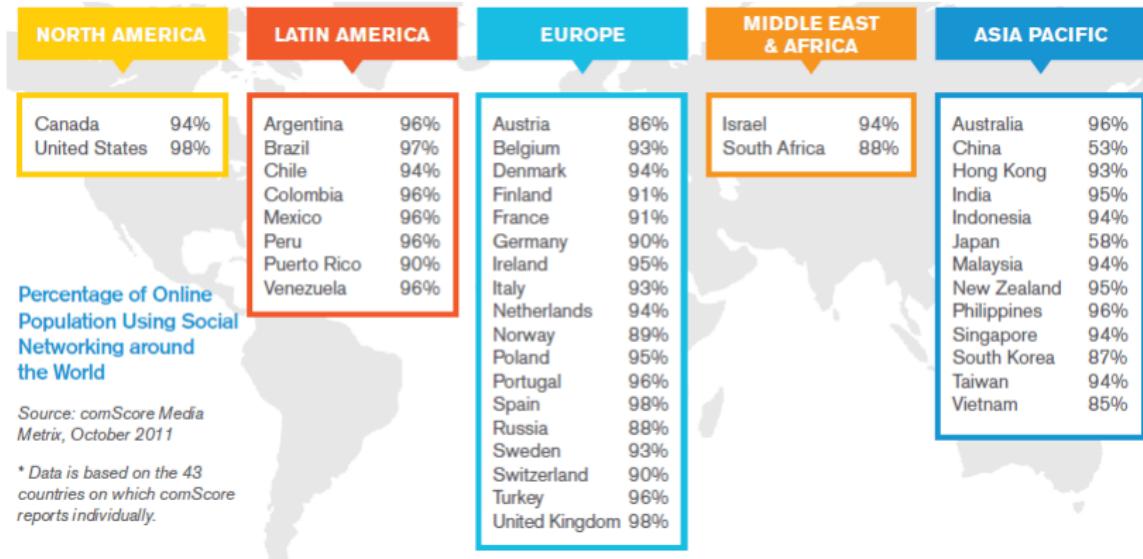
Top 10 Social-Networking Websites & Forums

by US Market Share of Visits (%)

March 2012

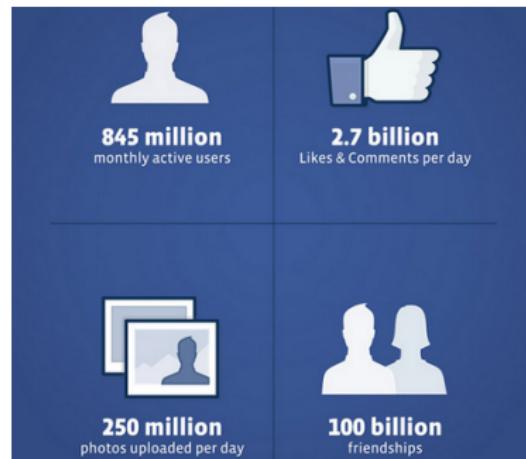


Популярность



Социальные данные

Особенности



Источники данных

- ▶ Слабо-структурированные источники реального времени

Размер данных

- ▶ Миллионы объектов с миллиардами связей

Модель данных

- ▶ Мультиграф из всех объектов и связей между ними

Социальные данные

Особенности

Типы данных

- ▶ Пользовательские аккаунты с профилями и связями
- ▶ Текстовый контент
- ▶ Мультимедийный контент
- ▶ Журналы активности пользователей

Особенности графовой структуры

- ▶ Безмасштабная сеть
- ▶ Число Данбара: средняя степень вершины равна 150
- ▶ Плотное ядро и разреженная периферия

Социальные данные

Сбор



Как скачать Facebook?

- ▶ Через API невозможно, нужно качать и парсить HTML
- ▶ Граф из 10^9 пользователей, **160** друзей в среднем
- ▶ $10^9 * 160 * 8$ (int64) = **1192 Гб** - только на хранение связей
- ▶ $10^9 * 100$ Кб (средний вес HTML-страницы) = **93 Тб** – трафик

Вывод: качать весь граф долго, дорого и непрактично

Решение: взять подграф (сэмпл), повторяющий свойства всей сети

Социальные данные

Сбор

Проблемы со стороны Facebook

- ▶ Обилие Javascript на стороне клиента
- ▶ Жёсткие настройки приватности
- ▶ Ограниченный доступ через API
- ▶ Блокировка (бан) по IP и аккаунту

Требования к кроулеру

- ▶ Эмуляция пользовательской сессии
- ▶ Скачивание и парсинг HTML
- ▶ Устойчивость к блокировке по IP и аккаунту
- ▶ Работа в параллельном/распределённом режиме

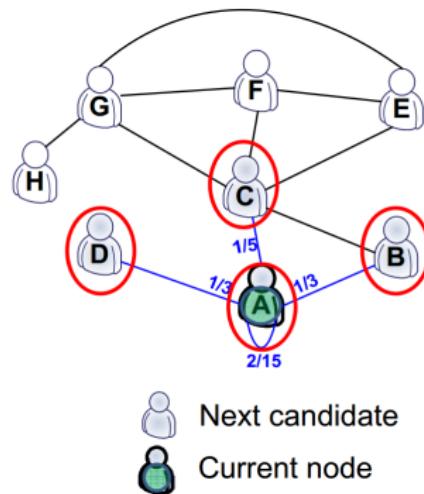
Требования к сэмплу

- ▶ Топология, близкая к оригинальной сети
- ▶ Консистентность

Социальные данные

Сбор

Metropolis-Hastings Random Walk

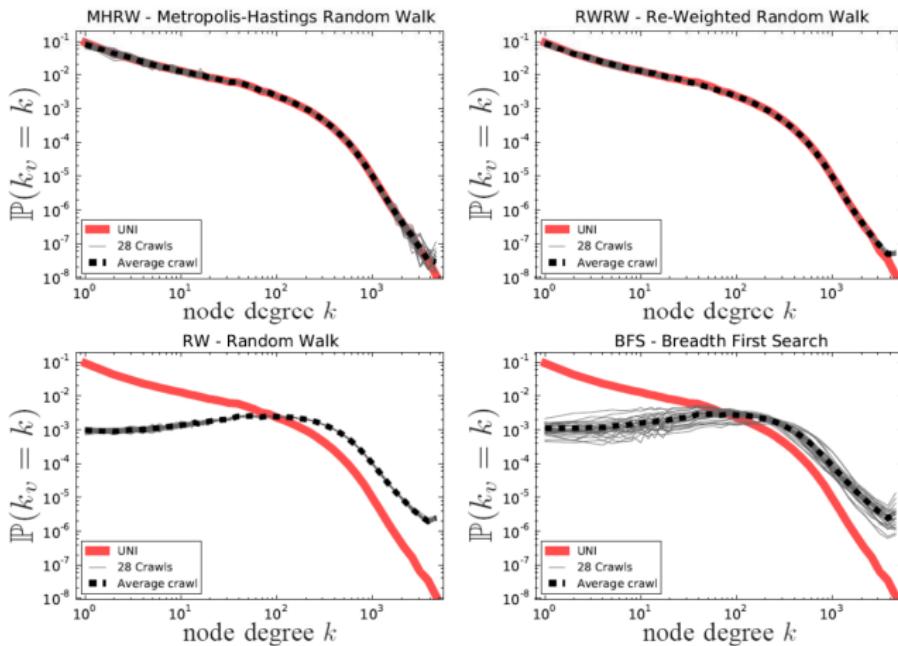


$$P_{v,w}^{MH} = \begin{cases} \min\left(\frac{1}{k_v}, \frac{1}{k_w}\right) & \text{if } w \text{ is a neighbor of } v, \\ 1 - \sum_{y \neq v} P_{v,y}^{MH} & \text{if } w = v, \\ 0 & \text{otherwise.} \end{cases}$$

Социальные данные

Сбор

Metropolis-Hastings Random Walk



Отдел информационных систем ИСП РАН

Направления работы по анализу социальных сетей

- ▶ Поиск сообществ пользователей
- ▶ Оценка информационного влияния
- ▶ Интеграция профилей пользователя
- ▶ Построение тематического профиля интересов пользователя или сообщества
- ▶ Определение скрытых атрибутов пользователей по текстам сообщений
- ▶ Определение эмоциональной окраски сообщений
- ▶ Рекомендация контента и пользователей
- ▶ Поиск событий в потоке пользовательских сообщений
- ▶ Генерация случайных графов со свойствами социальных сетей

Поиск глобальных сообществ пользователей



- ▶ Предсказание связей и атрибутов пользователей
- ▶ Оптимизация потоков данных
- ▶ Расчёт графовой близости
- ▶ Системы рекомендаций
- ▶ Аналитические приложения

Поиск глобальных сообществ пользователей

Требования

- ▶ Обязательные
 - ▶ Пересекающиеся сообщества
- ▶ Вторичные
 - ▶ Иерархическая структура сообществ
 - ▶ Поддержка взвешенных и направленных графов
 - ▶ Поддержка всех типов пользовательских данных

Производительность

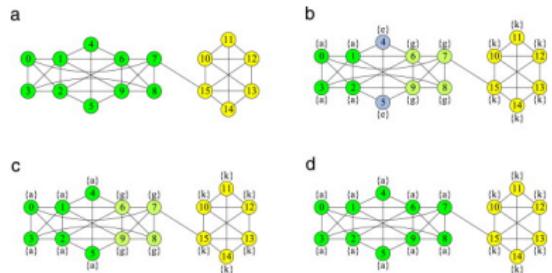
- ▶ Низкая вычислительная сложность
- ▶ Возможность распределённой реализации
- ▶ Хорошая масштабируемость

Nazar Buzun, Anton Korshunov Innovative Methods and Measures in Overlapping Community Detection

10th International Conference on Formal Concept Analysis May 2012, Leuven, Belgium

Поиск глобальных сообществ пользователей

Speaker-listener Label Propagation Algorithm (SLPA)



Алгоритм имитирует человеческое общение между парами индивидуумов:

1. Память каждого узла инициализируется уникальной меткой сообщества
 2. Затем итеративно повторяется последовательность шагов:
 - a. Выбирается "слушающий"узел
 - b. Каждая из вершин-соседей выбранного узла случайным образом выбирает метку с вероятностью, пропорциональной количеству меток данного типа в своей памяти, и посыпает выбранную метку "слушающему"узлу
 - c. "Слушающий"узел выбирает самую популярную из присланных ему меток и добавляет её в свою память
 3. Для поиска пересекающихся сообществ для каждой вершины выбираются самые популярные для них метки
 4. Для поиска непересекающихся сообществ для каждой вершины выбирается её наиболее популярная метка

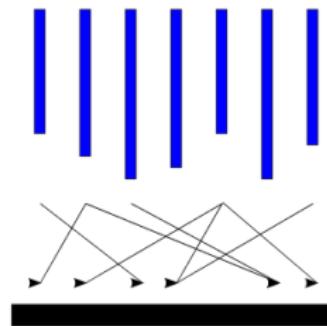
Google Pregel: обработка графовых данных на кластере

Модель Bulk Synchronous Parallel (BSP)

Вход Список вершин с атрибутами + список рёбер с атрибутами

Алгоритм Последовательность супершагов (итераций)

Выход Изменённый граф или рассчитанные статистики

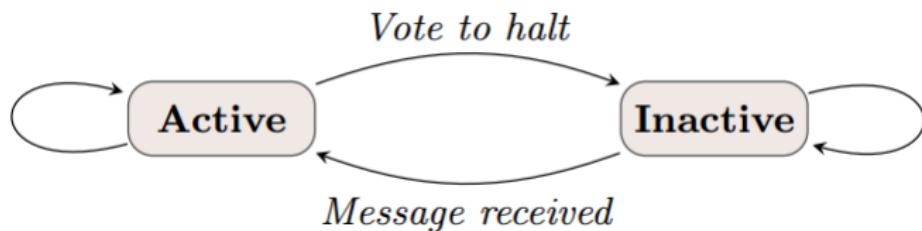


Супершаг - предопределённая функция, исполняемая всеми вершинами:

- ▶ приём сообщений с предыдущего супершага
- ▶ отправка сообщений (обычно соседним вершинам)
- ▶ изменение состояния вершины и/или соседей
- ▶ изменение структуры графа (добавление/удаление/изменение вершин/ребёр)

Google Pregel: обработка графовых данных на кластере

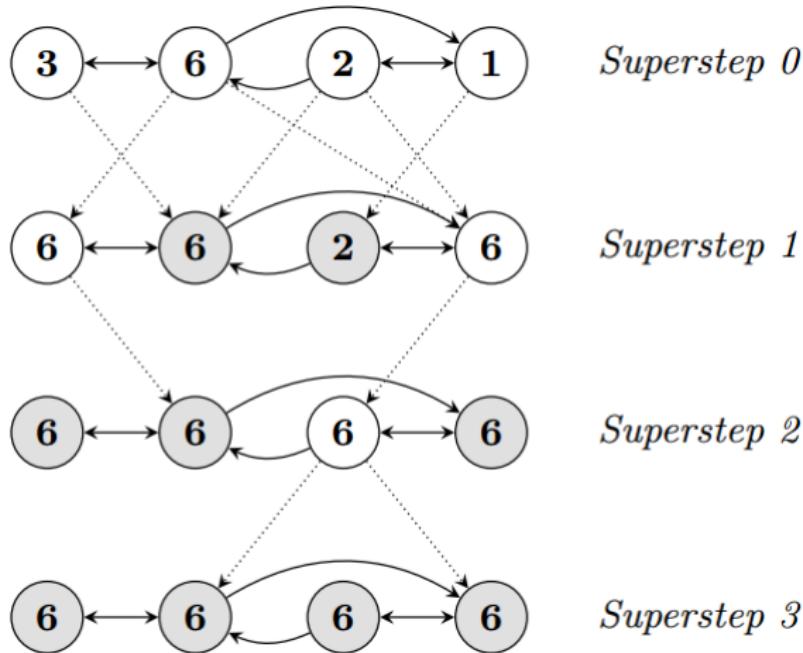
Модель вычислений



- ▶ На супершаге 0 все вершины активны
- ▶ Только активные вершины участвуют в очередном супершаге
 - ▶ Вершина становится неактивной путём голосования за завершение процесса вычислений
 - ▶ Реактивация возможна при получении сообщения
- ▶ Процесс останавливается единогласным решением всех вершин

Google Pregel: обработка графовых данных на кластере

Поиск максимального значения вершины



Реализация SLPA на Spark.Bagel



- ▶ Вершины графа распределяются по узлам кластера в RDD-структурах
- ▶ Ненаправленные рёбра трансформируются в двунаправленные
- ▶ На каждой итерации (супершаге) производится обмен метками между вершинами в виде сообщений (BSP)
- ▶ Вершины накапливают метки в памяти
- ▶ После завершения итераций, для получения окончательного разбиения применяется пост-процессинг памяти всех вершин
- ▶ Хорошая масштабируемость и высокое качество результатов

Алгоритм SLPA

Результаты оценки качества: LFR benchmark

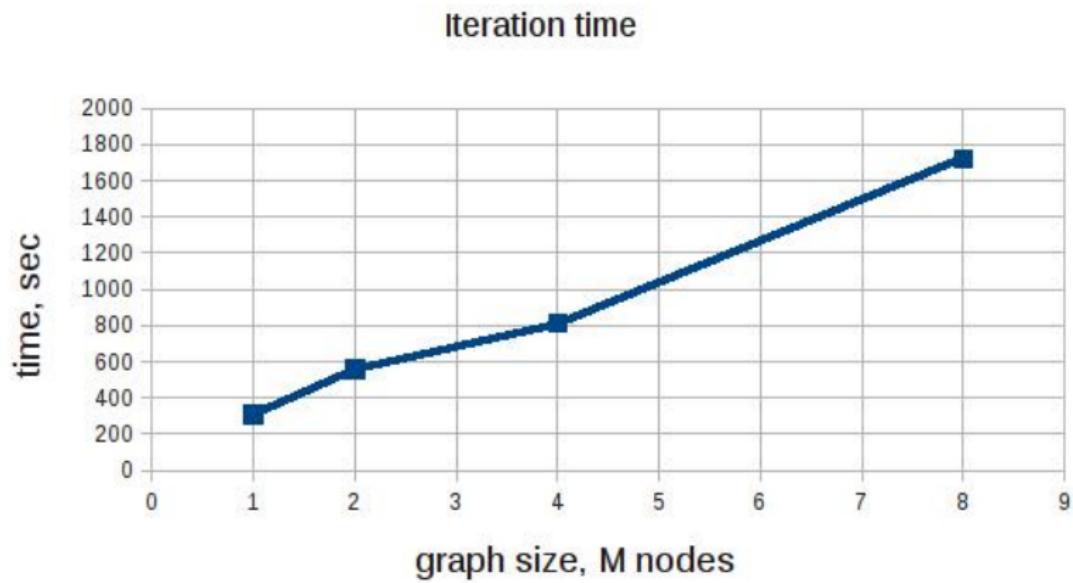
Взвешенный	Направленный	Пересек. сообщ.	NMI
-	-	+	0.82
-	-	-	0.97
-	+	+	0.95
-	+	-	0.09
+	-	+	0.95
+	-	-	1.00
+	+	+	0.75
+	+	-	0.98

Входные данные

- ▶ $|V| = 100\ 000$
- ▶ Доля вершин в пересекающихся сообществах = 0.1
- ▶ Количество пересекающихся сообществ на 1 вершину = 2

Алгоритм SLPA

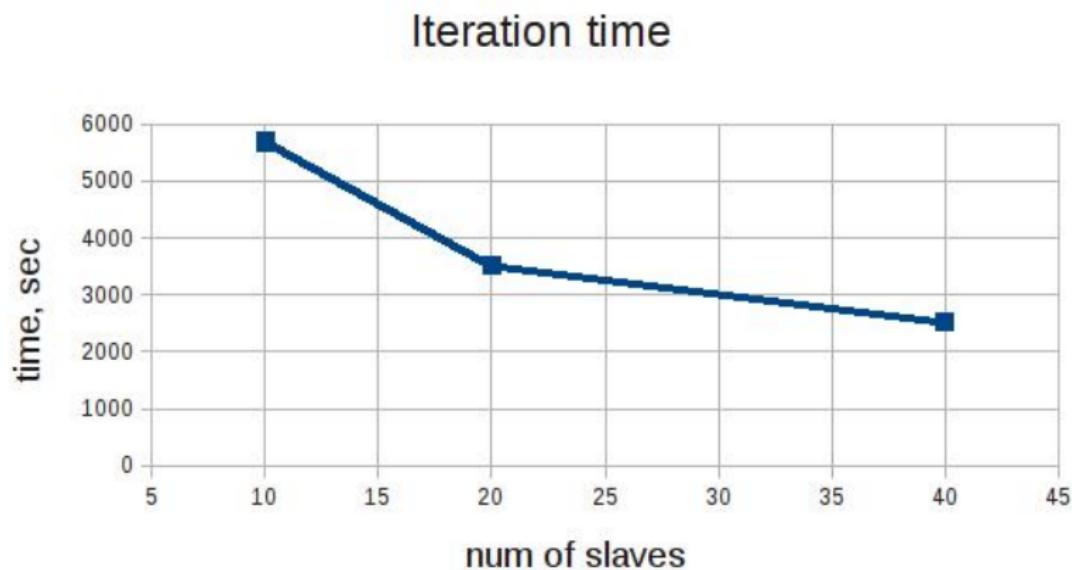
Масштабируемость от размера графа



20 узлов

Алгоритм SLPA

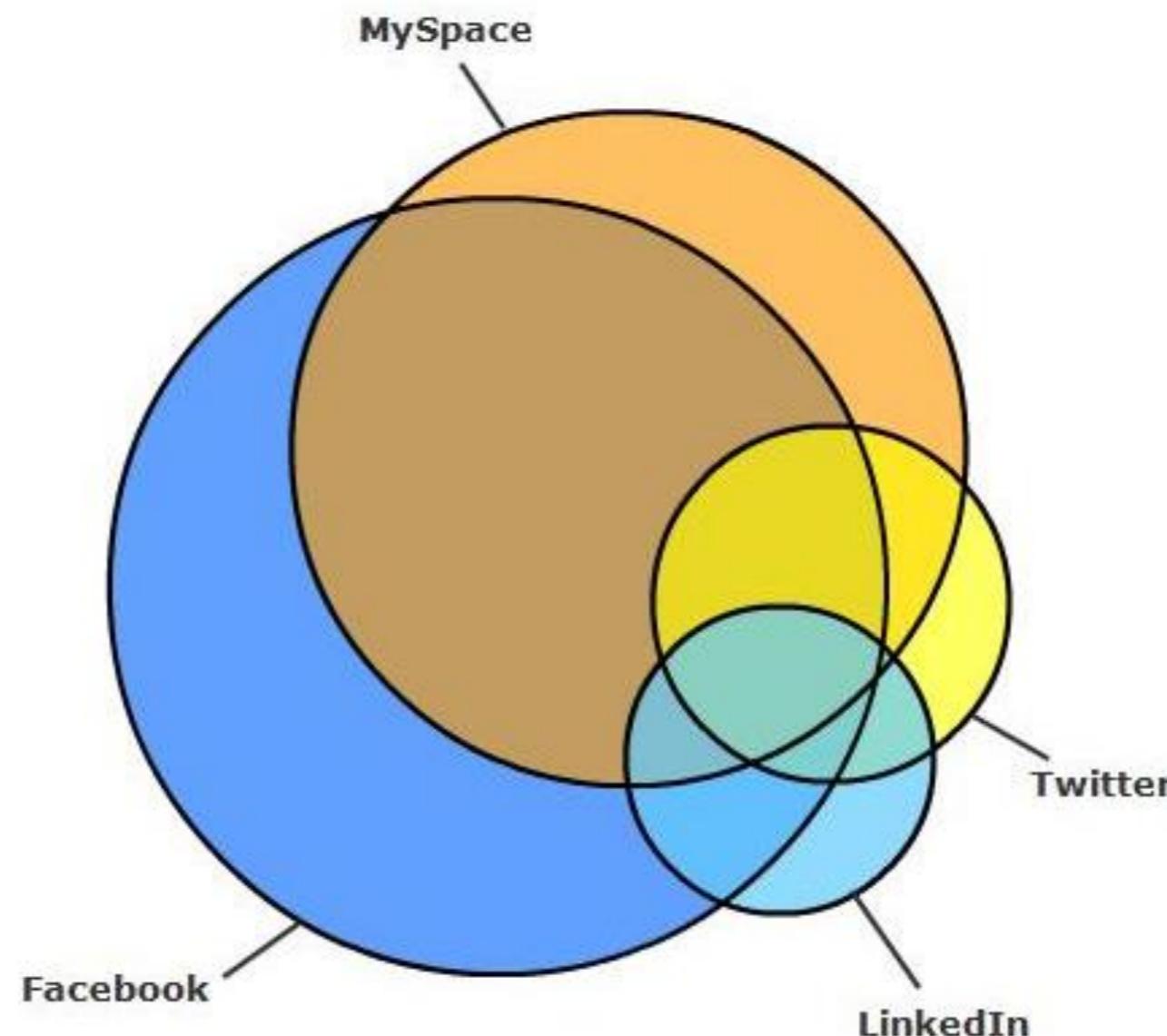
Масштабируемость от размера кластера



$$|V| = 4\ 254\ 507$$

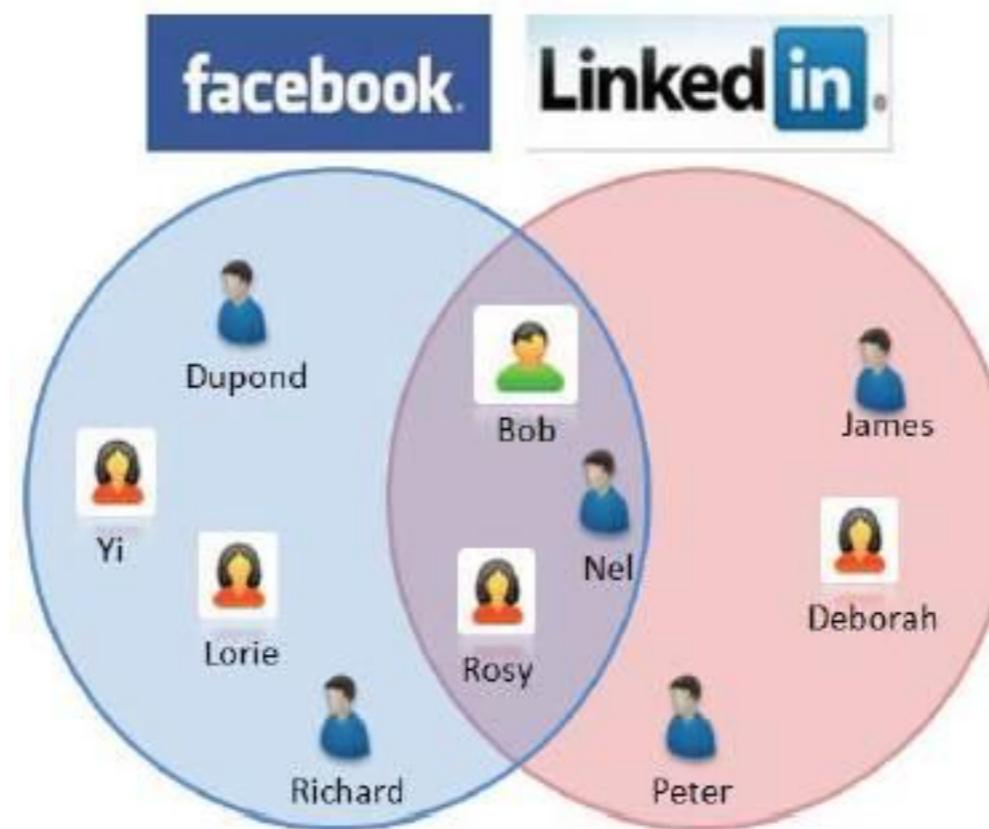
Интеграция профилей пользователя

SNS Usage Overlap



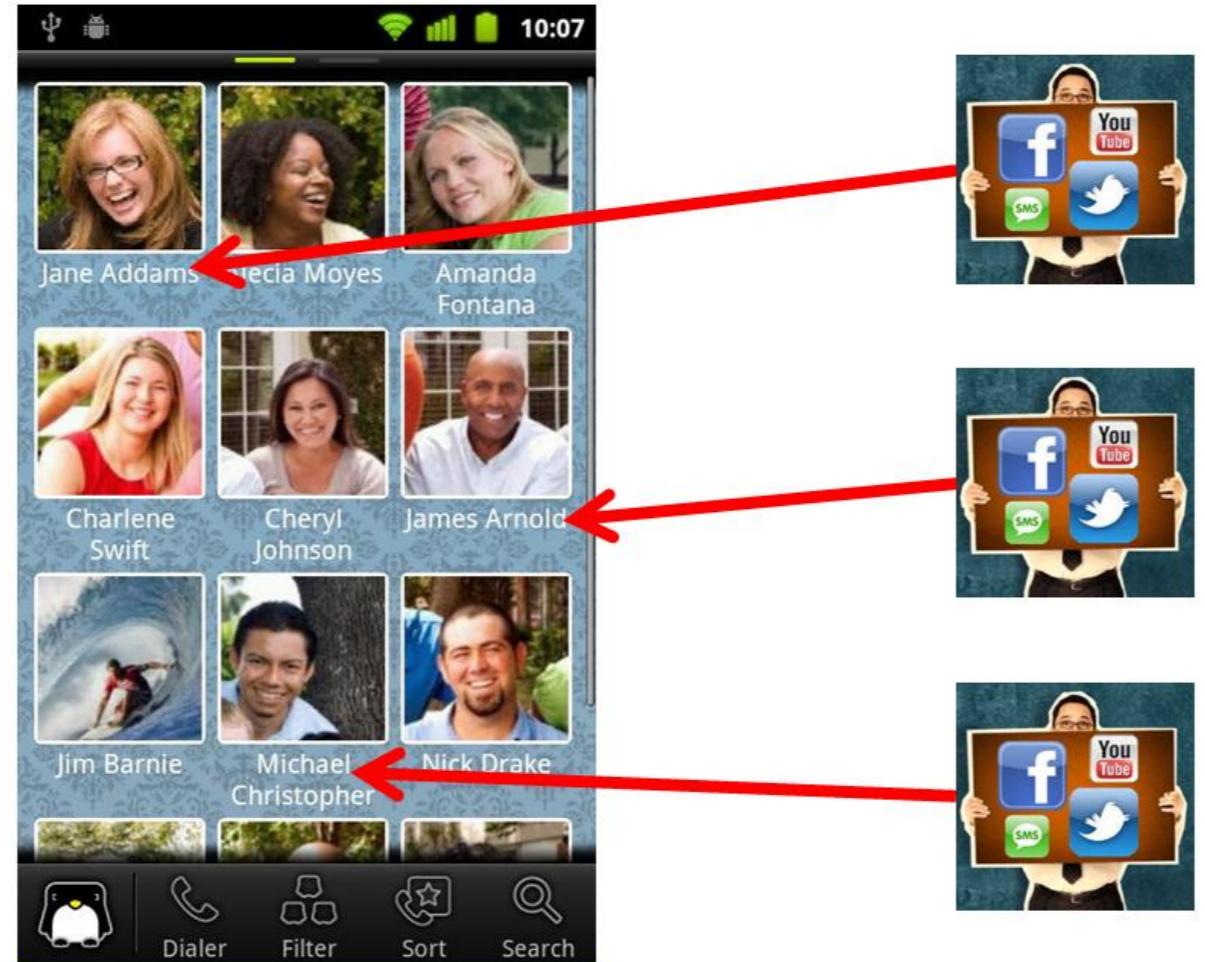
Source: Anderson Analytics 2009

Интеграция профилей пользователя



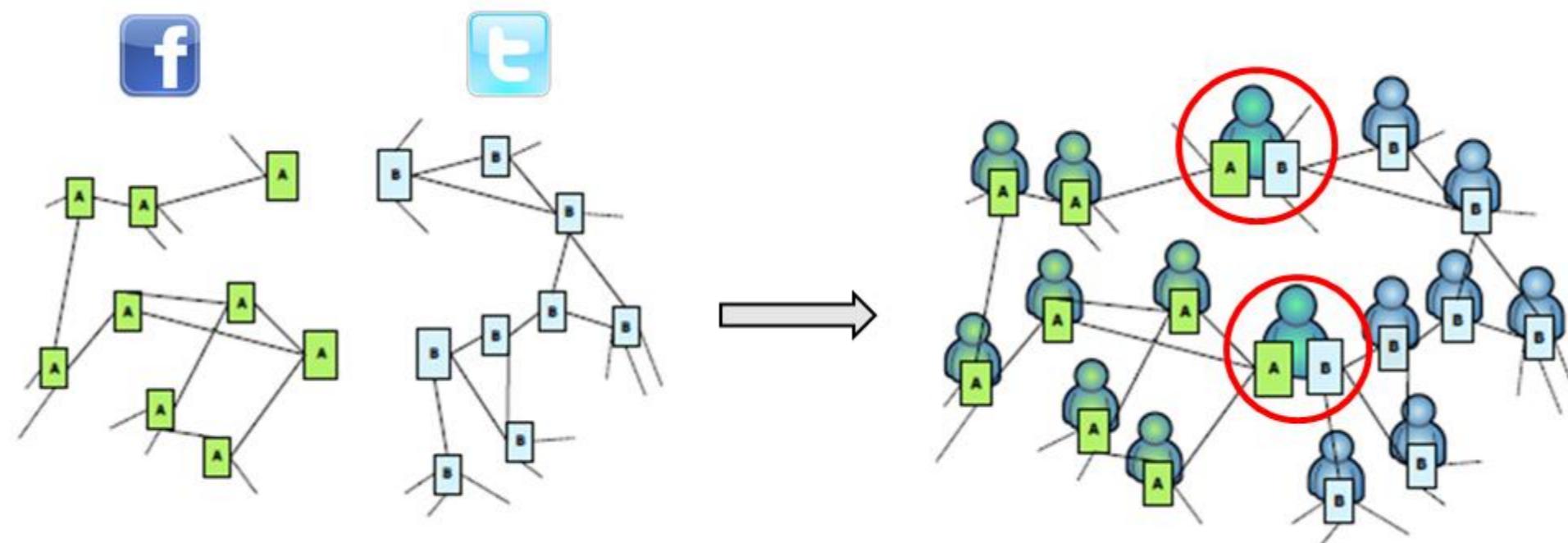
Как автоматически найти знакомых из одной сети в другой сети?

Интеграция профилей пользователя



Как автоматически объединить списки контактов?

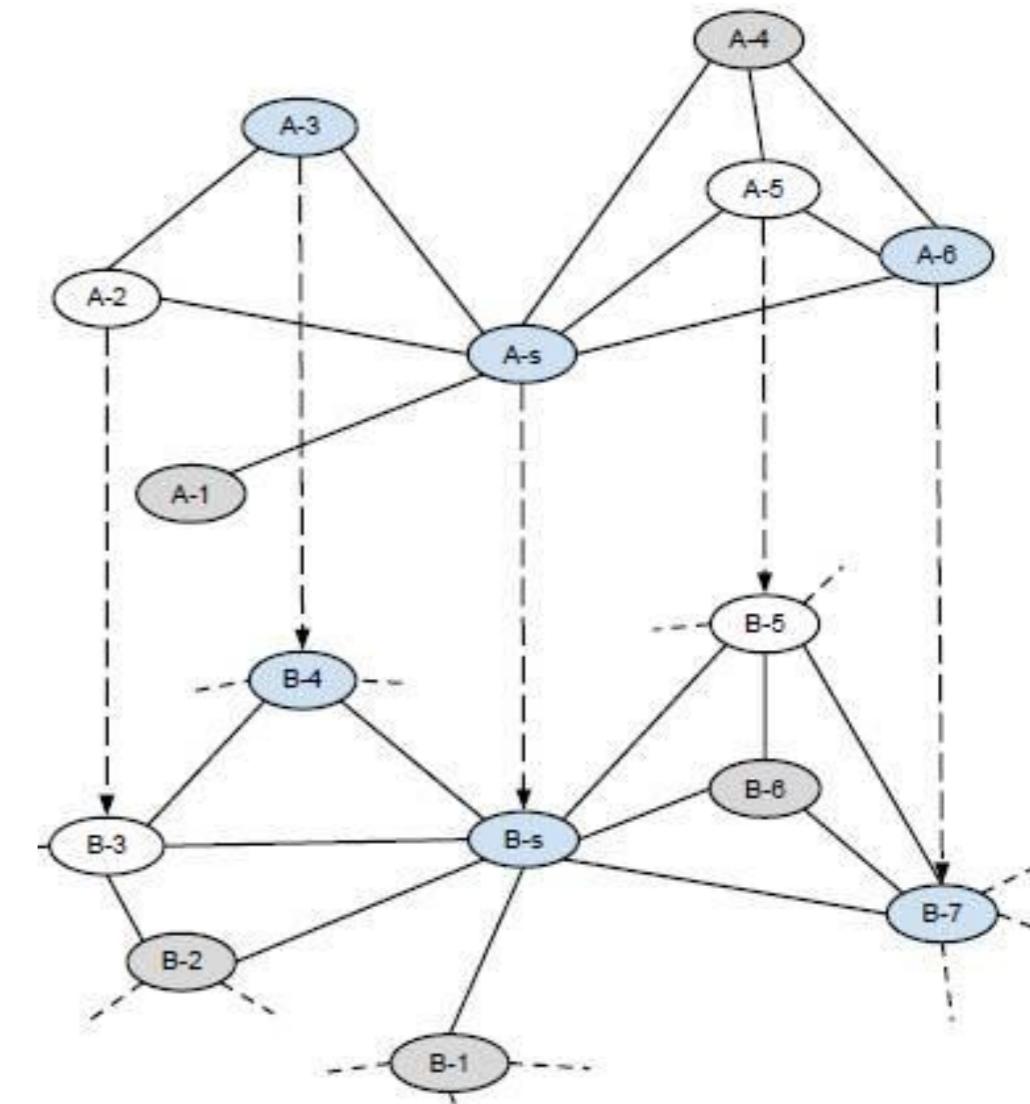
Интеграция профилей пользователя



Интеграция профилей пользователя

Решение

1. Построить модель условных
случайных полей
2. Найти пользователей с точным
совпадением по именам, добавить
их в модель
3. Рассчитать энергии рёбер с помощью
метрик строковой и графовой
близости
4. Найти оптимальную конфигурацию
совпадающих профилей путём
оптимизации общей энергии модели
5. Отфильтровать результаты



Интеграция профилей пользователя

algorithm	<i>R</i>	<i>P</i>	<i>F</i> ₁
agnostic to direction of projection			
Baseline 1 (weighted sum)	0.45	0.94	0.61
Baseline 2 (probability distance)	0.51	1.0	0.69
JLA, intersection, anonymized	0.6	1.0	0.76
JLA, intersection	0.66	0.99	0.79
Twitter → Facebook			
JLA, anonymized ($\Phi \equiv 0$)	0.62	1.0	0.77
JLA	0.79	1.0	0.89
Facebook → Twitter			
JLA, anonymized ($\Phi \equiv 0$)	0.61	1.0	0.76
JLA	0.8	1.0	0.89

Данные: 16 пар персональных сетей из Facebook и Twitter,
размеченных их владельцами

Определение скрытых атрибутов



Name: **Julia Stevens**
Age: **[empty]**
Gender: **female**
Relationship: **[empty]**
Location: **France**



Name: **Rob Fee**
Age: **666**
Gender: **female**
Relationship: **single**
Location: **U.S.**



Name: **Maria Zotova**
Age: **24**
Gender: **female**
Relationship: **married**
Location: **Moscow**

*атрибуты не
указаны*

ошибки

*дубликаты
профилей*

- Как определить пол, возраст, религию, статус отношений пользователя по текстам его сообщений?
- Кого больше среди фанатов игры/фильма/книги – мужчин или женщин, молодых или пожилых? Как использовать эти знания для рекомендации?

Определение скрытых атрибутов

Проблемы

- Лингвистические особенности сообщений Twitter
 - Максимум 140 символов
 - Неформальный стиль
 - Обилие сокращений, слэнга и микросинтаксиса
- Постоянное изменение интернет-слэнга
 - Нужно периодически заново обучать модель
- Ручная разметка данных занимает много времени и чревата ошибками
- Большое количество ретвитов и цитат
 - Доля авторского текста в микроблоге зачастую невелика

Определение скрытых атрибутов



Harry Bathurst 

@harrybathurst

18. To ammend the words of Toy Story, you have not got a friend
in! #bazinga

Brighton, UK - <http://facebook.com/harry.bathurst>



Harry Bathurst @harrybathurst

@Jedturner_1996 you see the video? youtube.com/watch?
v=8sGBld...

View media

11 Jul



Harry Bathurst @harrybathurst

@Jedturner_1996 what?

View conversation

11 Jul



Harry Bathurst @harrybathurst

Scorcher merked Dappyyyy haha!

Expand

11 Jul



Harry Bathurst @harrybathurst

You know I'm hot propertyyy

Expand

11 Jul

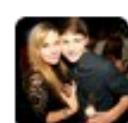


Harry Bathurst @harrybathurst

@peterquiffs funnyyyyy

Expand

11 Jul



Harry Bathurst @harrybathurst

Fed up of going to the gym on my own! #itsboring

Expand

11 Jul

Basic Info 	
Birthday	February 1, 1994
Sex	Male
Interested In	Women
Relationship	In a relationship with Emily Morgan
Status	
Anniversary	April 29, 2012
Languages	English
Religious Views	Fundamental Baptist
Political Views	Care Bears Dont care



Label: Gender	Label: Age	Label: Relationship
MALE	18	NON SINGLE

Определение скрытых атрибутов



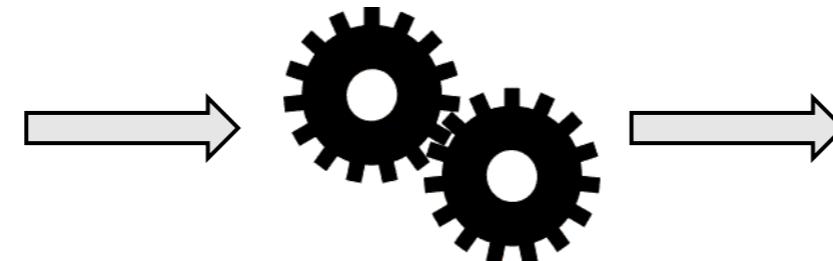
Решение

1. Извлечение бинарных признаков из твитов и профиля
2. Выбор наиболее информативных признаков для каждого класса
3. Обучение модели (алгоритм Passive-Aggressive)
4. Классификация

 **Rob Fee** @robfee 14h
Crossing my fingers that this is the season of Breaking Bad where Frankie Muniz finally catches his dad making meth.
[Expand](#)

 **Rob Fee** @robfee 23h
Kristen Stewart is what would happen if a Yawn took human form & got really into Snow Patrol.
[Expand](#)

 **Rob Fee** @robfee 23h
Check out this week's list of 20 Hilariously Inappropriate Tweets from Mandatory! mandatory.com/2012/07/13/thi...
[Expand](#)



Gender	Age	Relationship
MALE	<20	SINGLE

Определение скрытых атрибутов

Younger (≤ 20 years old)	Older (> 20 years old)
(('char', 6, u' <u>revisi</u> '), 0.15)	(('char_screen_name', 1, u' <u>8</u> '), 0.13)
(('token_uncased', 1, (u' <u>exam</u> ',)), 0.14)	(('token', 2, (u' <u>at</u> ', u' <u>work</u> ')), 0.13)
(('token', 2, (u' <u>:</u> ', u' <u>D</u> ')), 0.13)	(('token_uncased', 1, (u' <u>bar</u> ',)), 0.10)
(('char_uncased', 7, u' <u>bieber</u> '), 0.13)	(('char', 4, u' <u>le</u> :'), 0.10)
(('token_uncased', 1, (u' <u>prom</u> ',)), 0.13)	(('char', 4, u' <u>ern</u> '), 0.10)
(('token', 1, (u' <u>tomorrow</u> ',)), 0.13)	(('char', 2, u' <u>:</u> ('), 0.10)
(('token_uncased', 1, (u' <u>virgin</u> ',)), 0.12)	(('char_uncased', 7, u' <u>hool to</u> '), 0.09)
(('char', 3, u' <u>s</u> :'), 0.12)	(('char', 7, u' <u>season</u> '), 0.09)
(('char', 4, u' <u>nedi</u> '), 0.12)	(('token', 1, (u' <u>wine</u> ',)), 0.09)
(('char', 3, u' <u>:L</u> '), 0.12)	(('char_uncased', 7, u' <u>hit me</u> '), 0.09)
(('token', 2, (u' <u>≤</u> ', u' <u>3</u> ')), 0.12)	(('char', 3, u' <u>g</u> :'), 0.09)
(('char', 4, u' <u>Yo</u> '), 0.11)	(('char', 5, u' <u>House</u> '), 0.09)
(('char', 2, u' <u>(</u> '), 0.11)	(('char', 6, u' <u>ington</u> '), 0.09)
(('char', 5, u' <u>me?</u> '), 0.10)	(('token', 2, (u' <u>Nice</u> ', u' <u>!</u> ')), 0.08)
(('char', 5, u' <u>math</u> '), 0.10)	(('char_uncased', 4, u' <u>jew</u> '), 0.08)
(('char', 3, u' <u>KON</u> '), 0.10)	(('char', 5, u' <u>ompan</u> '), 0.08)
(('char', 7, u' <u>much i</u> '), 0.10)	(('char', 3, u' <u>Mo</u> '), 0.08)
(('char_uncased', 7, u' <u>favouri</u> '), 0.09)	(('char_uncased', 4, u' <u>o <3</u> '), 0.08)
(('char', 4, u' <u>Swa</u> '), 0.09)	(('char', 7, u' <u>real wo</u> '), 0.08)
(('char_uncased', 3, u' <u>t x</u> '), 0.09)	(('char_uncased', 4, u' <u>. y</u> '), 0.08)

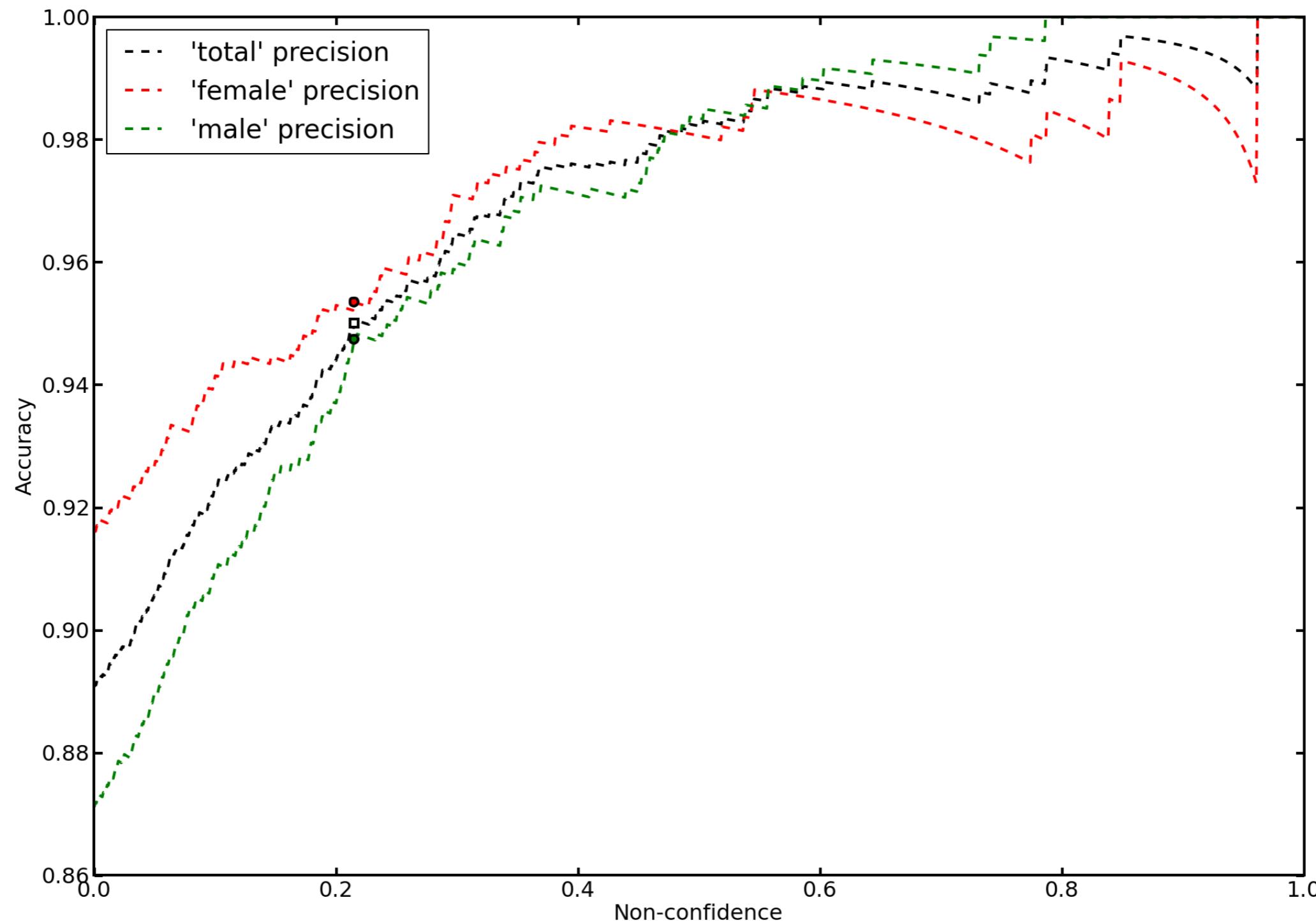
Определение скрытых атрибутов

Male	Female
(('token_name', 1, (u' <u>alex</u> ',)), 0.17)	(('char_name', 2, u' <u>a</u> '), 0.561)
(('char_name_uncased', 3, u' <u>an</u> '), 0.16)	(('char_name_uncased', 2, u' <u>i</u> '), 0.22)
(('char_name_uncased', 2, u' <u>k</u> '), 0.16)	(('char_name_uncased', 2, u' <u>e</u> '), 0.21)
(('char_name', 1, u' <u>_</u> '), 0.15)	(('char_name', 3, u' <u>ie</u> '), 0.18)
(('char_name', 2, u' <u>d</u> '), 0.14)	(('char_name_uncased', 2, u' <u>y</u> '), 0.17)
(('char_name', 2, u' <u>t</u> '), 0.14)	(('char_name', 1, u' <u>Y</u> '), 0.16)
(('char_name', 1, u' <u>k</u> '), 0.14)	(('char_name_uncased', 3, u' <u>rac</u> '), 0.16)
(('char_name', 2, u' <u>ev</u> '), 0.14)	(('char_name_uncased', 3, u' <u>jen</u> '), 0.14)
(('char_name_uncased', 3, u' <u>ust</u> '), 0.13)	(('char_name_uncased', 3, u' <u>ash</u> '), 0.13)
(('char_name', 4, u' <u>icha</u> '), 0.13)	(('char_name', 2, u' <u>hl</u> '), 0.13)
(('char_name', 1, u' <u>J</u> '), 0.13)	(('char_name_uncased', 4, u' <u>laur</u> '), 0.12)
(('char_name', 3, u' <u>ew</u> '), 0.12)	(('char_name_uncased', 2, u' <u>ka</u> '), 0.12)
(('char_name', 2, u' <u>m</u> '), 0.12)	(('char_name_uncased', 3, u' <u>ara</u> '), 0.12)
(('char_name_uncased', 3, u' <u>is</u> '), 0.12)	(('char_name', 2, u' <u>nn</u> '), 0.12)
(('char_name', 2, u' <u>am</u> '), 0.11)	(('char_name_uncased', 3, u' <u>ine</u> '), 0.12)
(('char_name_uncased', 2, u' <u>ry</u> '), 0.11)	(('char_name', 2, u' <u>Su</u> '), 0.11)
(('token_name', 1, (u' <u>chris</u> ',)), 0.11)	(('char', 3, u' <u>epy</u> '), 0.11)
(('char_name', 3, u' <u>ke</u> '), 0.11)	(('char_name', 3, u' <u>gan</u> '), 0.11)
(('char_name_uncased', 3, u' <u>wil</u> '), 0.11)	(('char_name', 2, u' <u>Am</u> '), 0.11)
(('char_name', 3, u' <u>And</u> '), 0.11)	(('char_name_uncased', 2, u' <u>sa</u> '), 0.11)

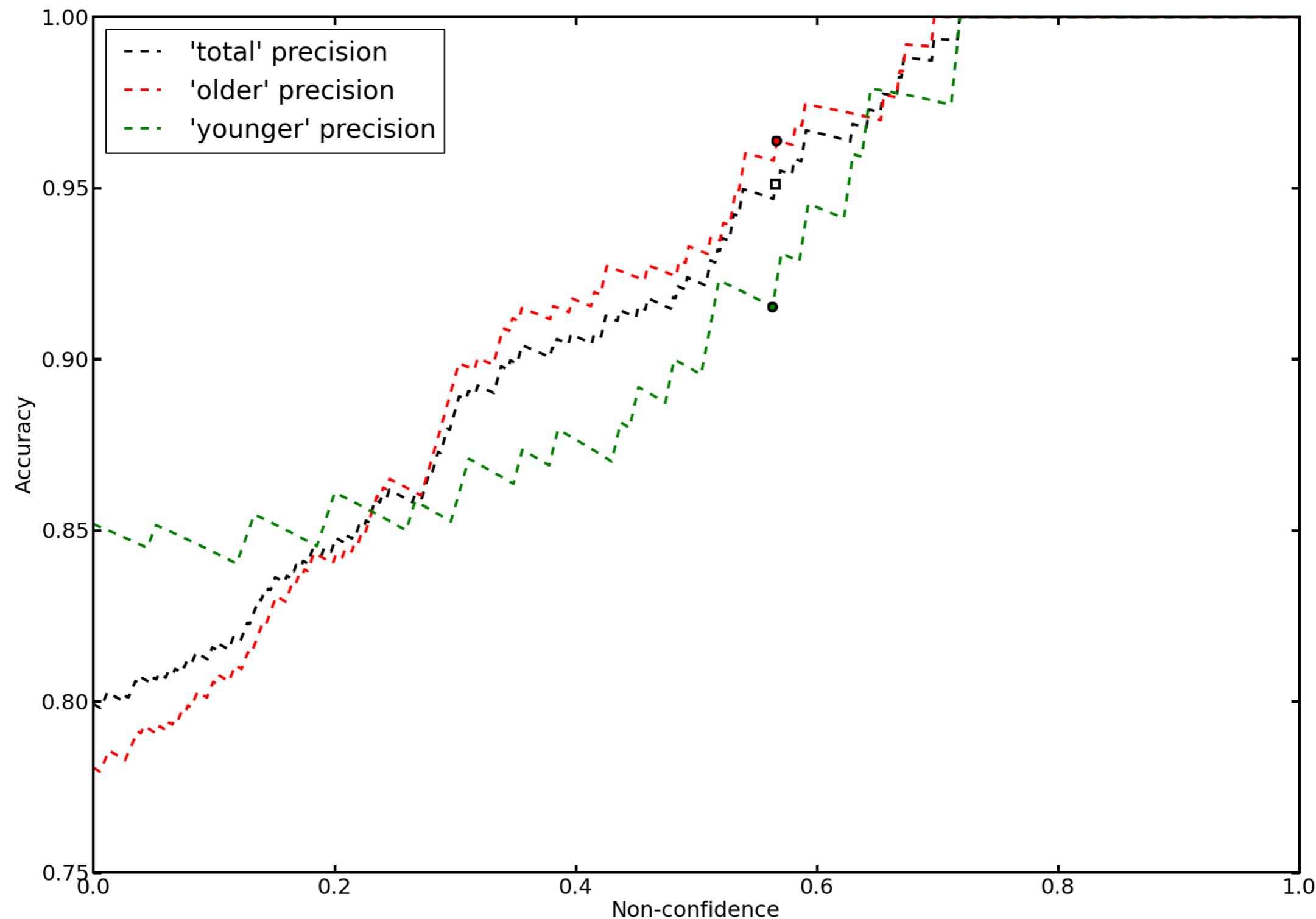
Определение скрытых атрибутов

Single	Non-single
(('char', 7, u' <u>Avenge</u> '), 0.21)	(('char', 2, u' <u>zh</u> '), 0.29)
(('char', 4, u' <u>bums</u> '), 0.19)	(('token', 2, (u' <u>my</u> ', u' <u>girlfriend</u> ')), 0.28)
(('char', 5, u' <u>d rel</u> '), 0.19)	(('char_uncased', 3, u' <u>bew</u> '), 0.23)
(('char', 5, u' <u>ys lo</u> '), 0.17)	(('char', 6, u' <u>y boyf</u> '), 0.22)
(('char_uncased', 4, u' <u>neat</u> '), 0.15)	(('token', 3, (u' <u>!</u> ', u' <u>!</u> ', u' <u>x</u> ')), 0.22)
(('char_uncased', 3, u' <u>d z</u> '), 0.15)	(('char', 5, u' <u>cool!</u> '), 0.22)
(('char_uncased', 3, u' <u>aja</u> '), 0.15)	(('char_name', 2, u' <u>el</u> '), 0.20)
(('char_uncased', 6, u' <u>e, can</u> '), 0.15)	(('char_uncased', 7, u' <u>burger</u> '), 0.20)
(('token', 2, (u' <u>:</u> ', u' <u>Thank</u> ')), 0.15)	(('char_uncased', 4, u' <u>poop</u> '), 0.20)
(('char', 7, u' <u>golden</u> '), 0.14)	(('char_uncased', 6, u' <u>done.</u> '), 0.19)
(('char_uncased', 4, u' <u>tip</u> '), 0.14)	(('token', 2, (u' <u>mate</u> ', u' <u>_</u> ')), 0.19)
(('char', 3, u' <u>RAG</u> '), 0.14)	(('char_uncased', 5, u' <u>roche</u> '), 0.19)
(('char', 2, u' <u>\$5</u> '), 0.14)	(('char_uncased', 4, u' <u>e? t</u> '), 0.18)
(('char', 6, u' <u>haha w</u> '), 0.14)	(('char_uncased', 5, u' <u>baby</u> '), 0.18)
(('char_uncased', 3, u' <u>kur</u> '), 0.14)	(('char_uncased', 2, u' <u>8z</u> '), 0.18)
(('char', 6, u' <u>locke</u> '), 0.14)	(('char_uncased', 7, u' <u>tell s</u> '), 0.18)
(('token_uncased', 3, (u' <u>in</u> ', u' <u>the</u> ', u' <u>future</u> ')), 0.14)	(('char', 5, u' <u>#ilo</u> '), 0.17)
(('char_uncased', 7, u' <u>dancers</u> '), 0.14)	(('char', 6, u' <u>e you!</u> '), 0.17)
(('token_uncased', 3, (u' <u>heading</u> ', u' <u>back</u> ', u' <u>to</u> ')), 0.14)	(('token', 1, (u' <u>shift</u> ',)), 0.17)
(('char', 5, u' <u>ine..</u> '), 0.14)	(('char_uncased', 5, u' <u>neil</u> '), 0.17)

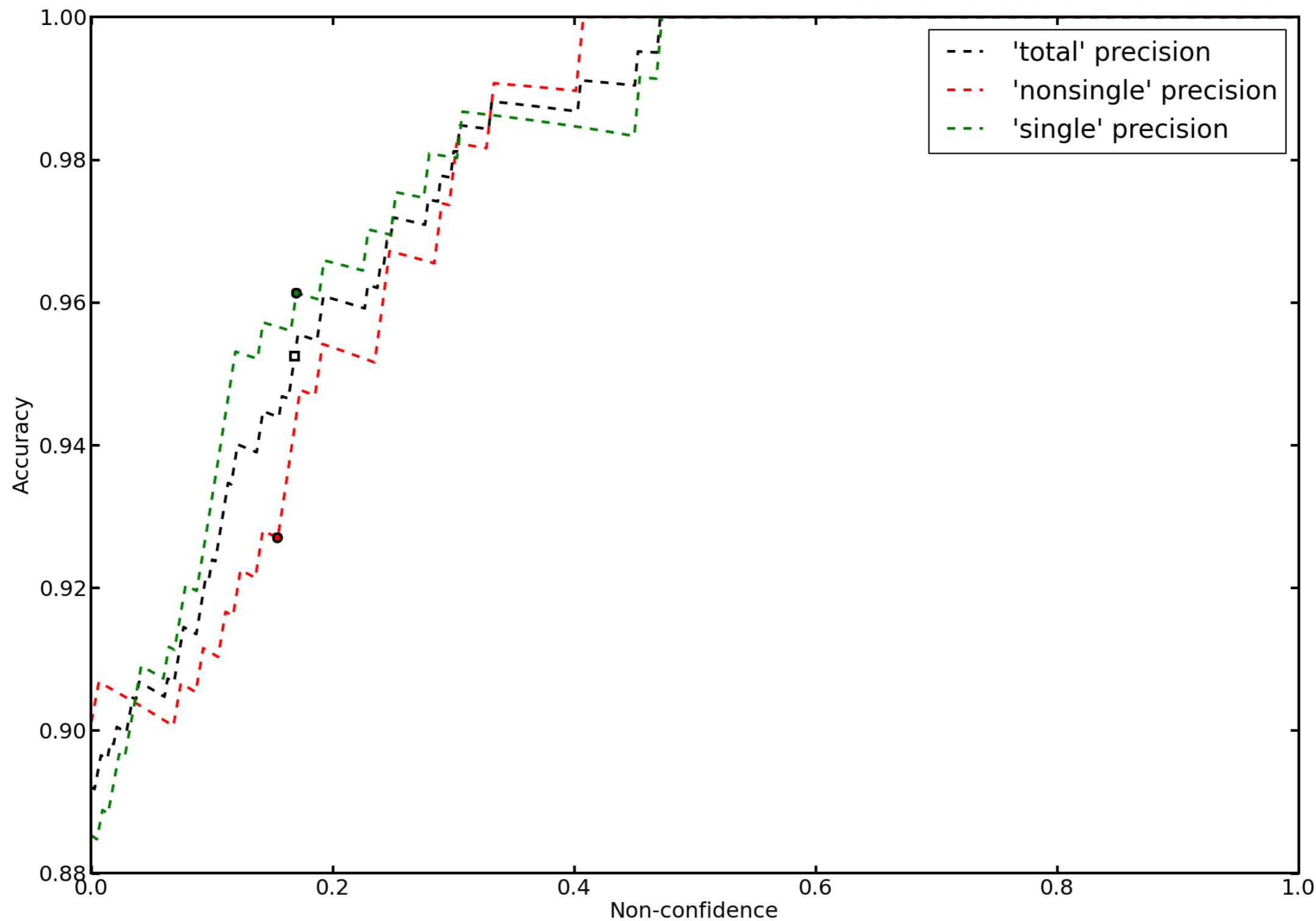
Определение скрытых атрибутов



Определение скрытых атрибутов



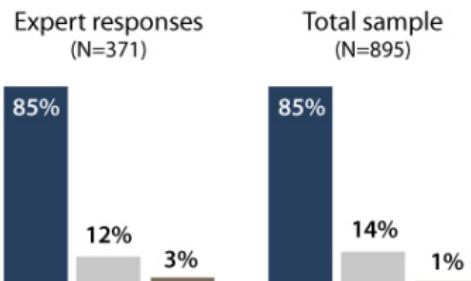
Определение скрытых атрибутов



Что дальше?

Will social relations get better?

Tension pair on the future of social relations

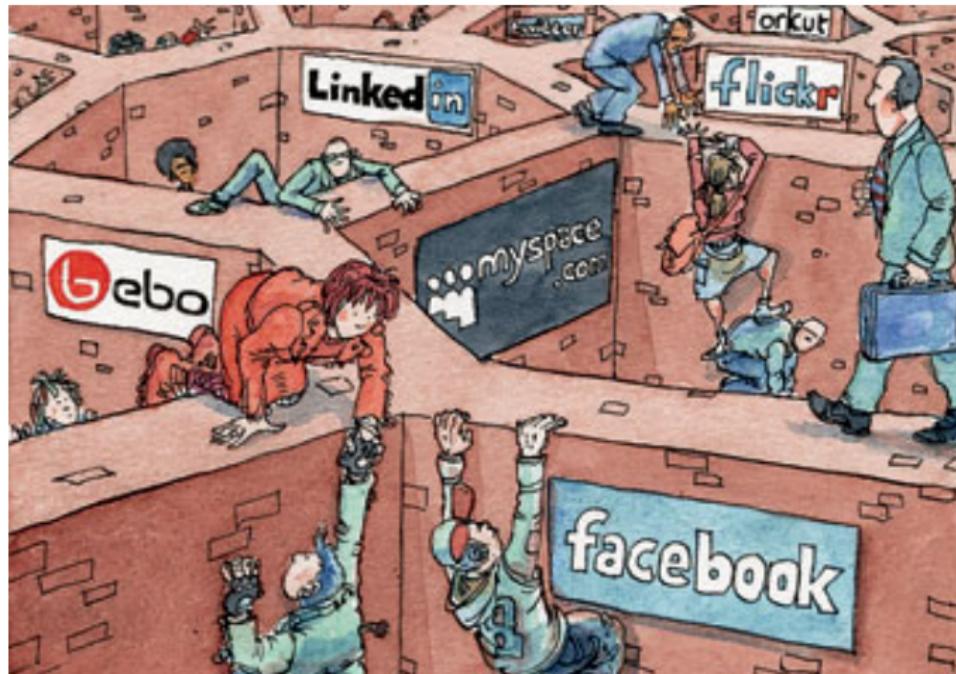


- In 2020, when I look at the big picture and consider my personal friendships, marriage, and other relationships, I see that **the internet has mostly been a positive force** on my social world. And this will only grow more true in the future.
- In 2020, when I look at the big picture and consider my personal friendships, marriage, and other relationships, I see that **the internet has mostly been a negative force** on my social world. And this will only grow more true in the future.

■ Did not respond

Что дальше?

Децентрализация



Что дальше?

Децентрализация

▼ Dan Brickley    



This is you

Basic Information

danbri homepage
my.opera homepage
danbri.livejournal weblog
my.opera weblog

Openid  <http://danbri.livejournal.com/>

Nickname [danbri](#)
[danbri](#)

Acquaintances

- Aaron Swartz
- Amy van der Hiel
- Art Barstow
- [Benjamin Joffe](#)
- [Charles McCathieNevile](#)
- Damian Steer
- [Dan Connolly](#)
- Dave Beckett
- [Dean Jackson](#)
- [Edd Dumbill](#)
- [Eddie Lopez](#)
- [Eric Miller](#)
- [Eva Méndez](#)
- [Gregory J. Rosmaita](#)

 Jan Grant

- Jim Ley
- Joe Brickley

Tip: Do you have [web ID](#)?
[Make or set A Web ID](#)

SeeAlso ► <http://foaf.qdos.com/reverse/?path=http://danbri.org/foaf.rdf#danbri>

Что дальше?

Новые сервисы

- ▶ Создание новых средств для самовыражения, а также обмена знаниями и опытом
- ▶ Более глубокое понимание особенностей человеческого поведения
- ▶ Социальное экспериментирование



Оценивание ответов



Деятельность → кредиты



Обмен рецептами

Вопросы?

Social Media Explained



A vintage-style illustration of a smiling man and woman eating bacon.

-  I'm eating bacon
-  I like bacon
-  I have skills including eating bacon
-  This is where I eat bacon
-  Watch me eat my bacon
-  Here's a vintage photo of my bacon
-  Here's a recipe with bacon
-  I work for Google and eat bacon
-  I'm listening to music about bacon

©COREY SMITH coreysmith.ws