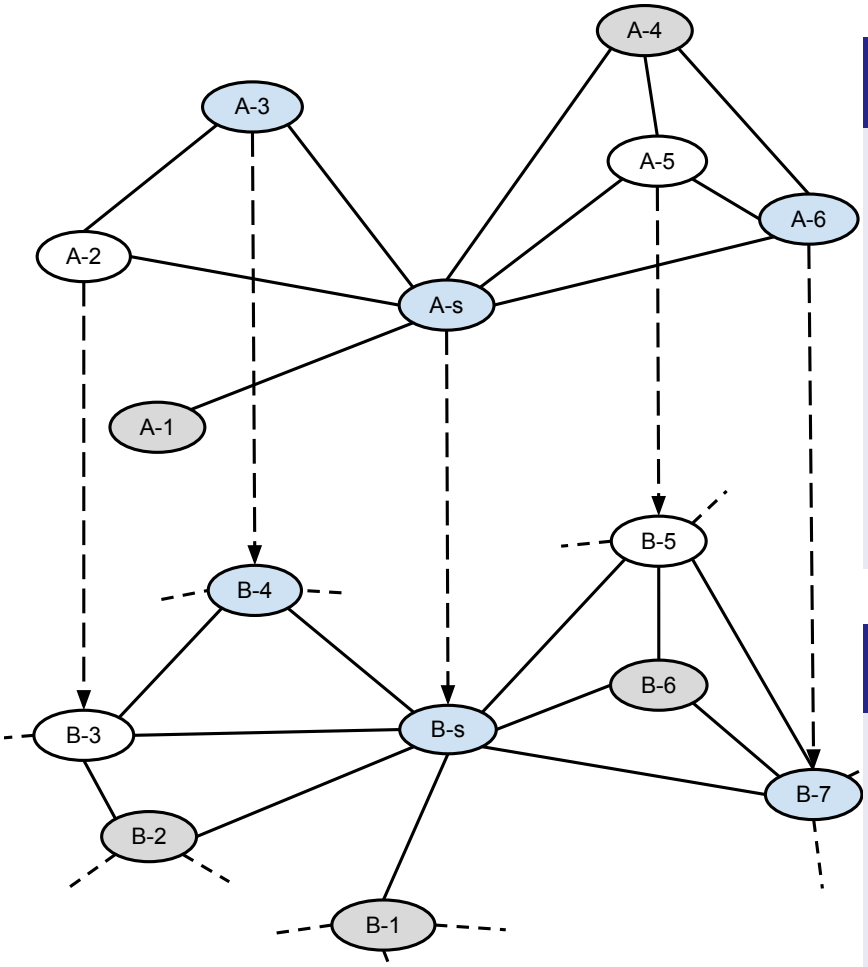
Joint Link-Attribute User Identity Resolution In Online Social Networks

Сергей Бартунов, Антон Коршунов

ИСП РАН

22 февраля 2012 г.

Постановка задачи



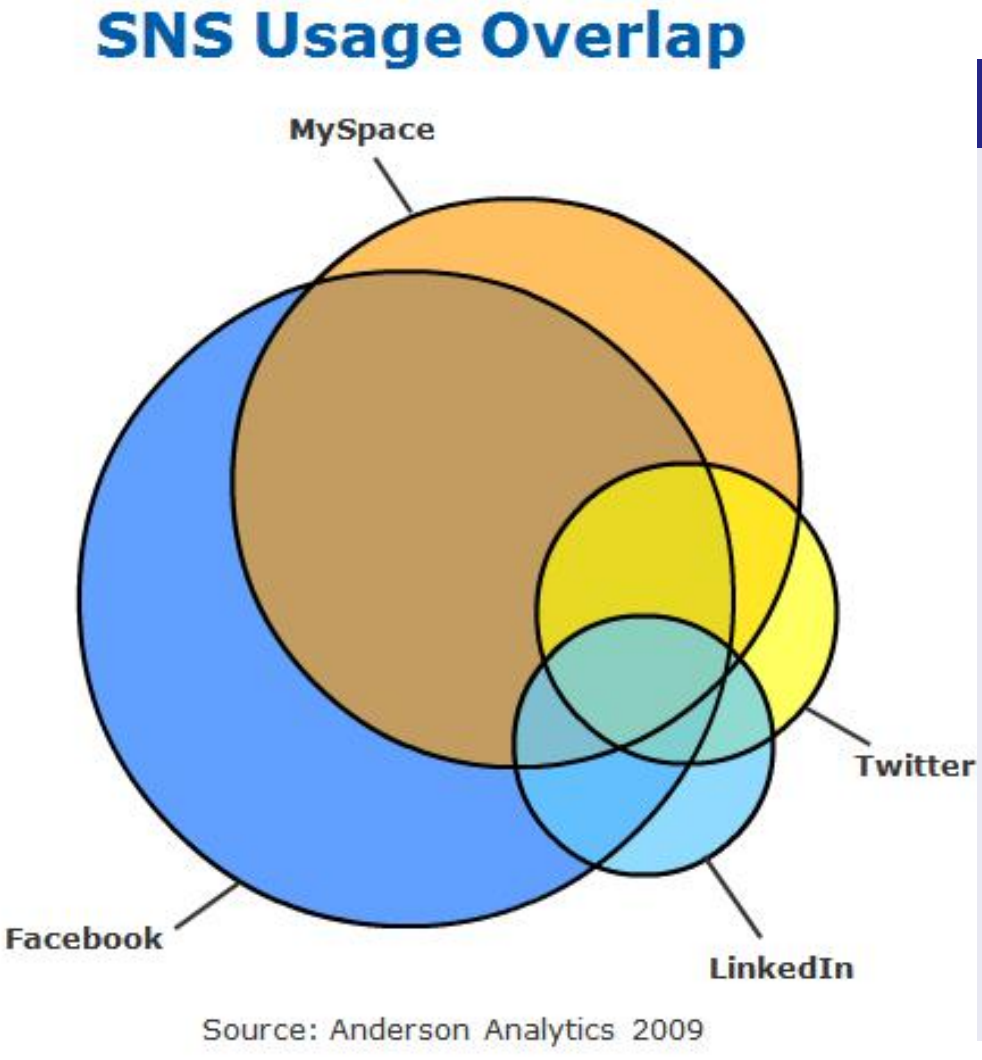
Данные

- Два социальных графа $< A, B >$:
  - Поля профилей (имя, адрес, день рождения и т.д.)
  - Социальные связи (друзья, подписка, ...)

Задача

Найти как можно больше пар профилей $(v, u) \mid v \in A, u \in B$, принадлежащих одному человеку

Мотивация



Мотивация

- Многие используют социальные сети и имеют несколько аккаунтов
- Есть множество нишевых социальных сетей
- Социальная информация может сильно помощь во многих приложениях, нужен более полный социальный граф
- Объединение контактов

Профили в социальных сетях

Sergey Bartunov

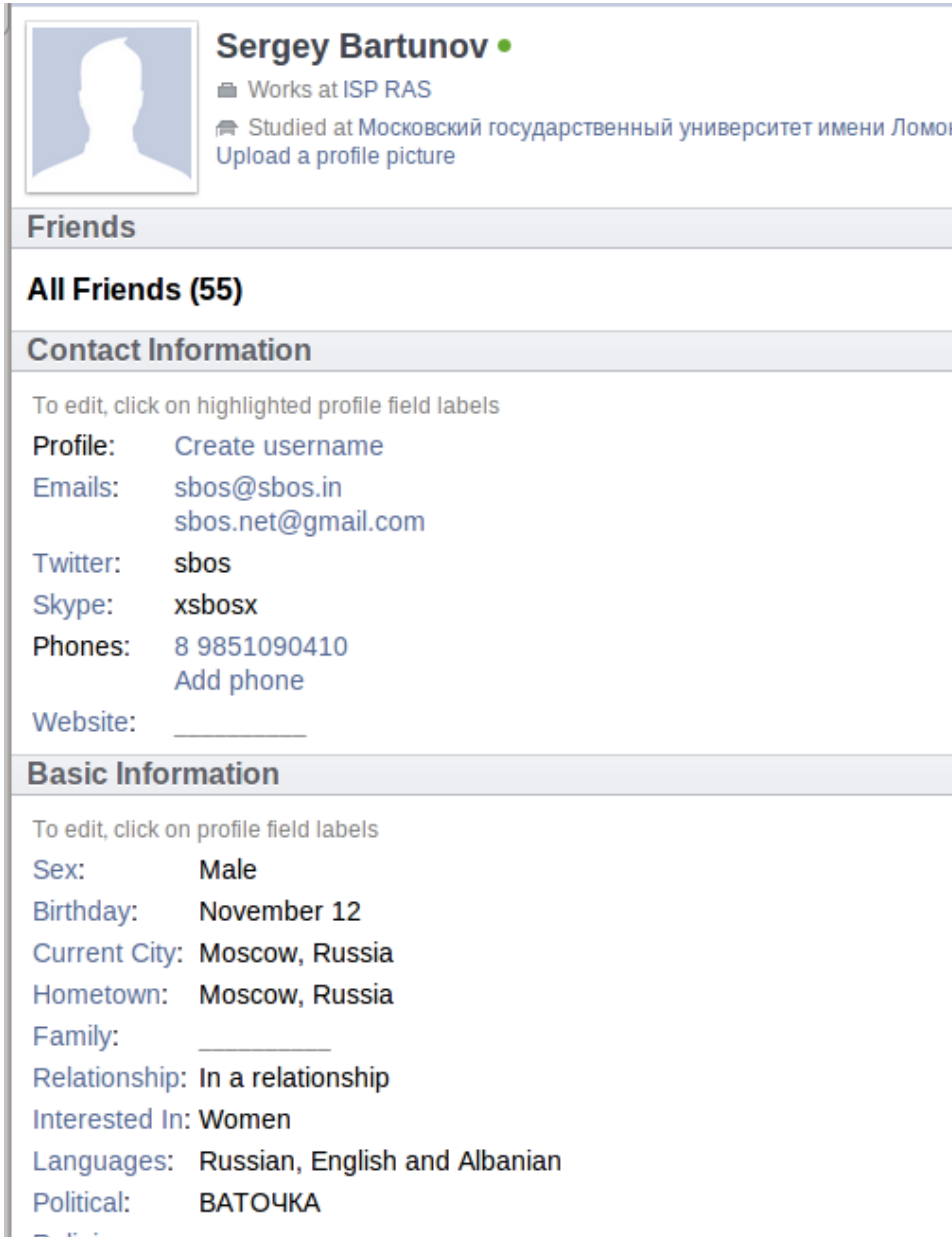
@sbos

Беспечный спецездок! Зайцы съедят меня, когда я стану травой

Moscow

Профили в социальных сетях

- Некоторые содержат множество данных (Facebook)
- Некоторые практически никакой (Twitter)



Sergey Bartunov ●

Works at ISP RAS

Studied at Московский государственный университет имени Ломо

Upload a profile picture

Friends

All Friends (55)

Contact Information

To edit, click on highlighted profile field labels

Profile:	Create username
Emails:	sbos@sbos.in sbos.net@gmail.com
Twitter:	sbos
Skype:	xsbosx
Phones:	8 9851090410 Add phone
Website:	_____

Basic Information

To edit, click on profile field labels

Sex:	Male
Birthday:	November 12
Current City:	Moscow, Russia
Hometown:	Moscow, Russia
Family:	_____
Relationship:	In a relationship
Interested In:	Women
Languages:	Russian, English and Albanian
Political:	ВАТОЧКА

Существующие подходы

Attribute-based UIR

- Поля профилей сравниваются с помощью функций нечеткого сравнения строк
- Результаты взвешиваются, суммируются и сравниваются с пороговым значением

Недостатки

- Не всегда пользователи аккуратно заполняют поля профилей или держат их в актуальном состоянии
- Иногда люди придумывают ники, а не вводят реальные имена
- Одинаковые имена не всегда означают одного владельца
- Профили вообще не всегда доступны из-за приватности

Существующие подходы

Сравнение частично сопоставленных списков контактов

- Сначала профили сопоставляются по полям
- Затем в качестве дополнительной информации привлекается показатель близости сопоставленных друзей

Похожесть списков контактов

- $J(L_v, L_u) = \frac{|L_v \cap L_u|}{|L_v \cup L_u|}$
- $cos(L_v, L_u) = \frac{|L_v \cap L_u|}{\sqrt{|L_v||L_u|}}$
- $dice(L_v, L_u) = \frac{2|L_v \cap L_u|}{|L_v| + |L_u|}$

Существующие подходы

Сравнение частично сопоставленных списков контактов

- Сначала профили сопоставляются по полям
- Затем в качестве дополнительной информации привлекается показатель близости сопоставленных друзей

Недостатки

- Техника опирается на ненадежные поля профилей
- Такой показатель близости очевидно „предвзят” (biased)

Существующие подходы в близких областях

Разрешение сущностей

- Даны записи в базе данных
- Надо определить все записи, относящиеся к одному объекту реального мира (не обязательно описывающие его)

Применяется CRF

- Наблюдаемые переменные - узлы-факты
- Скрытые переменные - узлы-утверждения (о том, к какому объекту относится запись)
- Утверждения связаны с фактами и между собой
- Ищется MAP-конфигурация, которая отражает истинность утверждений

Существующие подходы в близких областях

Преимущества

- Задачи взаимозависимы
- Работает лучше чем то, что было раньше

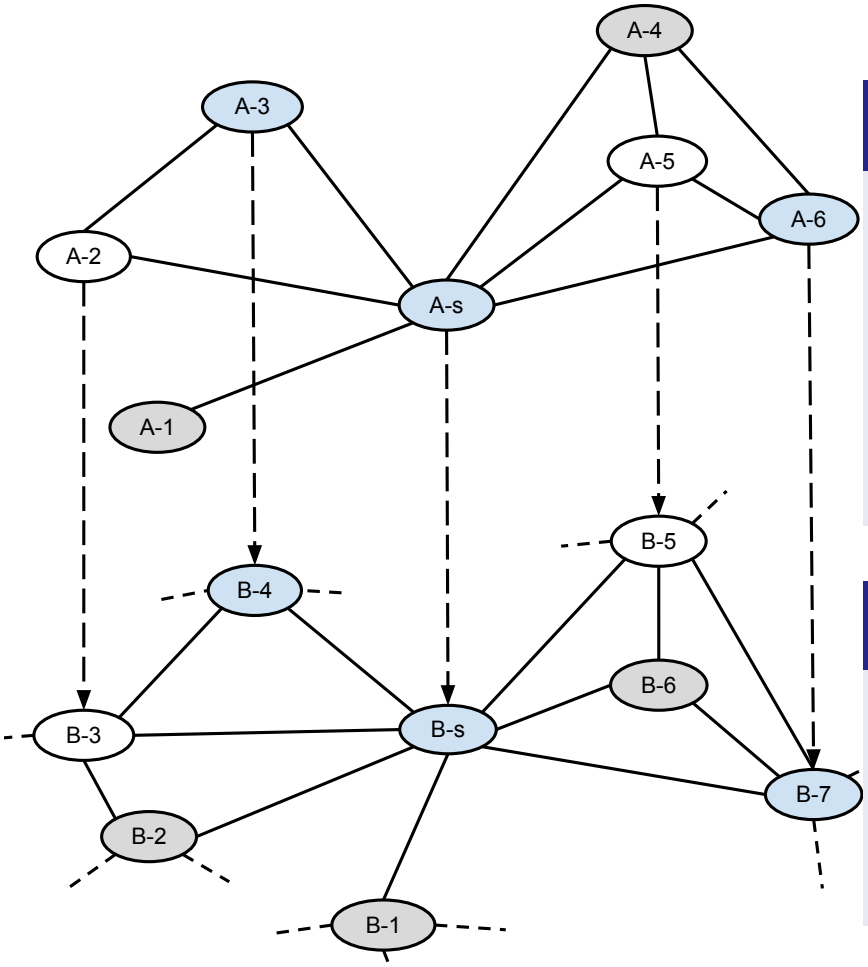
Недостатки

- Чрезмерно большая и подробная модель
- Никогда не использовалась близость в графе

Литература

- P. Singla, P. Domingos. *Entity Resolution with Markov Logic*. (ICDM'06).
- P. Singla, P. Domingos. *Multi-relational Record Linkage*. KDD Workshop on Multi-Relational Data Mining, 2004.

Обозначения



Локальная перспектива

- Центральный профиль
- Все профили, с ним связанные
- Списки контактов этих профилей

Задача

Найти оптимальное отображение

$\mu : V(A) \to V(B) \cup \mathbf{N}$

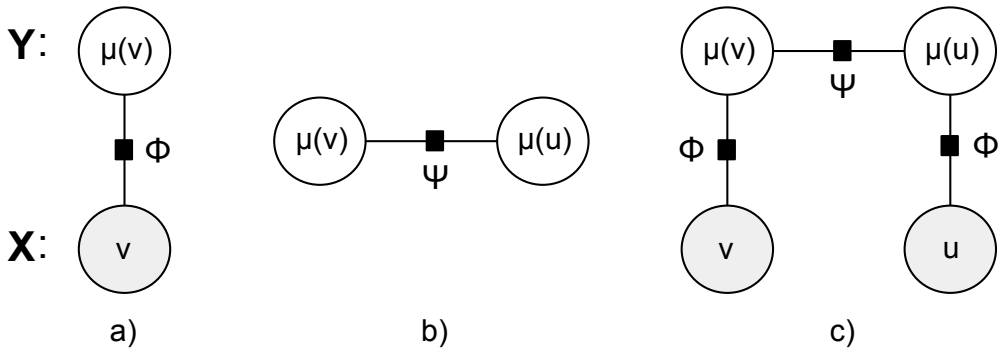
N - нейтральная проекция

Joint Link-Attribute Model

Основные положения

- Необходимо использовать, как информацию из профилей, так и социальные связи
- Задачи выбора проекций $\mu(v)$ и $\mu(u)$ для связанных вершин v и u взаимосвязаны
- Если v и u связаны в графе A, то их проекции в графе B должны быть близки друг к другу

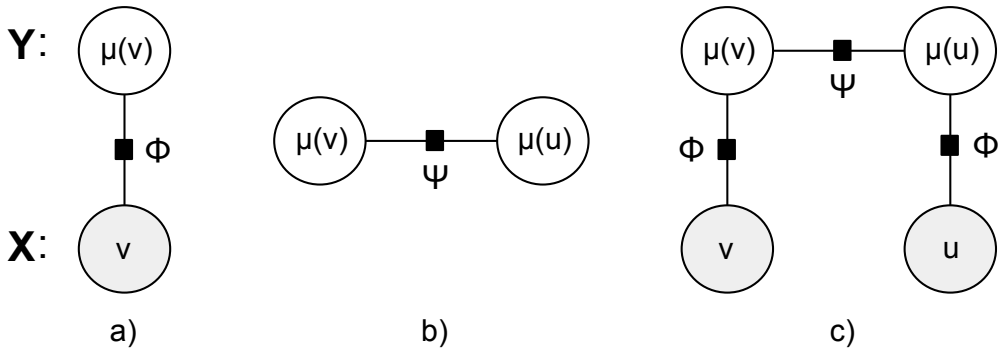
Joint Link-Attribute Model



Вероятностная модель

- Для каждой вершины v в графе A:
  - Наблюдаемая переменная $\mathbf{x}_v$ - профиль v
  - Скрытая переменная $\mathbf{y}_v = \mu(v)$ - проекция профиля v в графе B
- Переменные $\mathbf{x}_v$ и $\mathbf{y}_v$ связаны фактором Φ
- Переменные $\mathbf{y}_v$ и $\mathbf{y}_u$ связаны фактором $\Psi \Leftrightarrow (v, u) \in E(A)$

Joint Link-Attribute Model

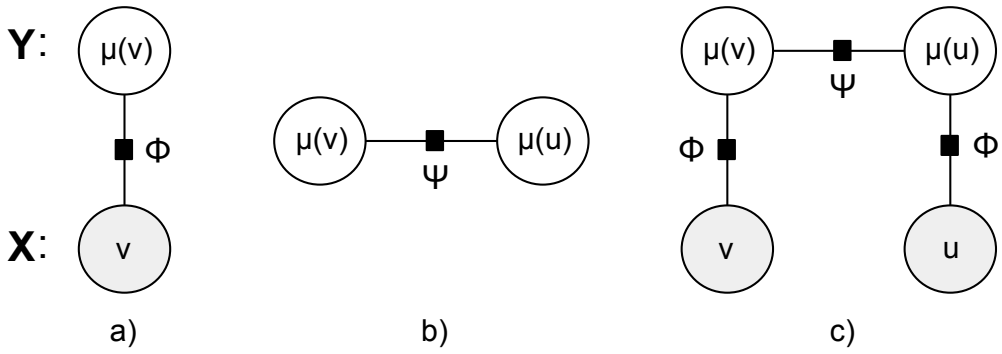


Энергия модели

$$E(\mathbf{Y}|\mathbf{X}) = \sum_{v \in V(A)} \Phi(\mathbf{y}_v|\mathbf{x}_v) + \sum_{(v,u) \in E(A)} \Psi(\mathbf{y}_v, \mathbf{y}_u)$$

- $\Phi(\mathbf{y}_v|\mathbf{x}_v) \sim$ profile-distance$(v, \mu(v))$ - (не)похожесть профиля на свою проекцию
- $\Psi(\mathbf{y}_v, \mathbf{y}_u) \sim$ network-distance$(\mu(v), \mu(u))$ - расстояние между проекциями в графе B

Joint Link-Attribute Model



Joint nature

$$E(\mathbf{Y}|\mathbf{X}) = \sum_{v \in V(A)} \Phi(\mathbf{y}_v|\mathbf{x}_v) + \sum_{(v,u) \in E(A)} \Psi(\mathbf{y}_v, \mathbf{y}_u)$$

a) Если не рассматривать социальные связи, то $\Psi \equiv 0$, и модель вырождается до тривиальной системы

b) Если недоступна информация о полях профилей, то $\Phi \equiv 0$, и решается задача деанонимизации

c) Вся информация доступна

Заранее известные проекции

Заранее известные проекции

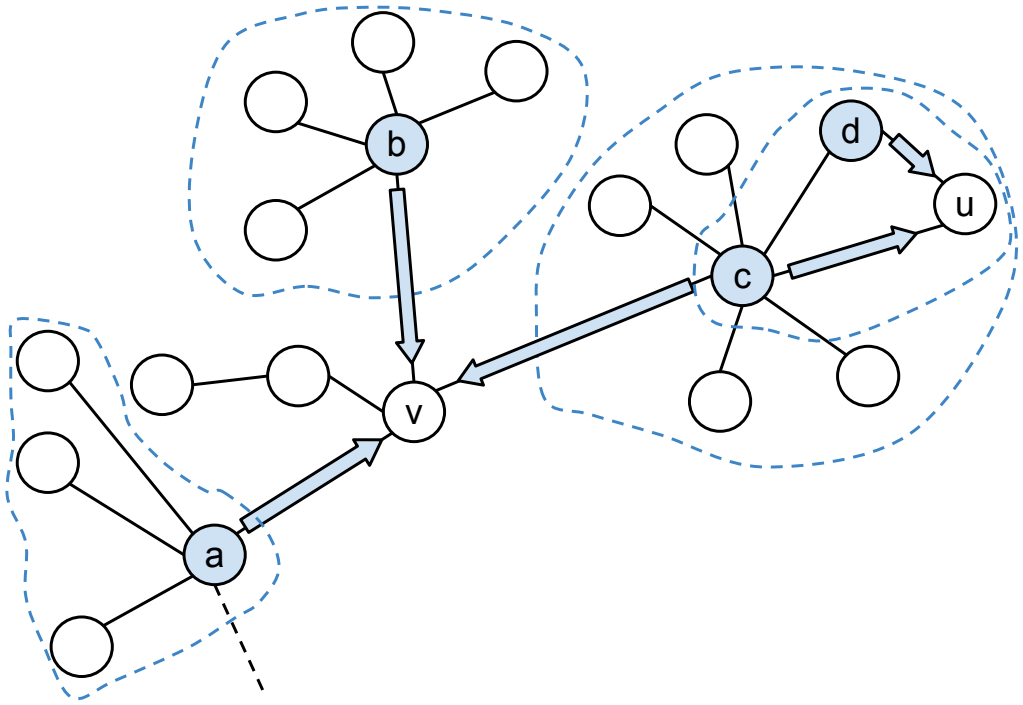
- Для некоторых вершин проекции могут быть известны заранее
- Некоторые можно считать таковыми, если $\text{profile-distance}(v, \text{pr}(v)) \leq \Delta$

Модификация энергии

Для каждой вершины v с заранее известной проекцией $\text{anchor}(v)$ запрещаются другие проекции:

$$\Phi(\mathbf{y}_v \mid \mathbf{x}_v) = \infty$$
$$\Psi(\mathbf{y}_v, \mathbf{y}_u) = \Psi(\mathbf{y}_u, \mathbf{y}_v) = \infty \qquad \text{если } \mathbf{y}_v \neq \text{anchor}(v)\ \forall u$$

Распространение информации



Заранее известные проекции

- Улучшают результаты работы
- Позволяют декомпозировать задачу

Похожесть профилей в Twitter и Facebook

Схема сравнения

Facebook	Twitter	Функция сравнения
Name	Name	VMN
	Screen name	Screen Name measure
Website	URL	URL measure

Функции сравнения

- Screen Name проверяет не совпалили ли явно ник и имя
- URL measure проверяет не указал ли явно пользователь второй профиль
- VMN позаимствована из Vosecky et. al., *User identification across multiple social networks*, 2009.

Похожесть профилей в Twitter и Facebook

Вектор похожести

- К каждой паре полей применяется соответствующая функция сравнения
- В результате получается *вектор похожести* $V(v, \mu(v))$

Функция (не)похожести

- Вектор $V(v, \mu(v))$ можно использовать как набор признаков для обучения бинарного классификатора
- profile-distance$(v, \mu(v)) = \mathrm{P}(\text{разные люди} | V(v, \mu(v))$

Похожесть профилей в Twitter и Facebook

Сравнение классификаторов

алгоритм	полнота	точность	F_1
Naive Bayes	**0.862**	0.308	0.453
C4.5	0.569	0.86	0.685
C4.5 с MultiBoosting	0.669	**0.879**	**0.76**

Выводы

- В целом, это работает
- Ни один классификатор не смог идеально „объяснить” принадлежность профиля

Расстояние в графе между проекциями

Функция расстояния

$$\text{network-distance}(v, u) = 1 - \frac{2 \cdot w(L_v \cap L_u)}{w(L_v) + w(L_u)} \quad v, u \in B$$

- Используется коэффициент Дайса (вообще говоря, это не расстояние)
- L_v - список контактов профиля v
- $w(L) = |L|$ - вес множества контактов (можно по-разному взвешивать)
- Быстро считается и отражает „марковность”

Взвешивание энергий

- profile-distance $\in [0, 1]$
- network-distance $\in [0, 1]$
- У каждой вершины унарная энергия одна
- Бинарных энергий много

Баланс между энергиями

- $\Phi(\mathbf{x}_v|\mathbf{y}_v) = \alpha(v) \cdot \text{profile-distance}(v, \mu(v))$
- $\alpha(v) = log(degree(v))$ - балансирующий коэффициент

Ограничения на бинарную энергию

Запрет на одинаково сопоставленные соседние профили

- Очевидно, что
  $\text{network-distance}(v, v) \geq \text{network-distance}(v, u) \quad \forall v, u$
- Но нам не нужен одинаково размеченный граф
- Поэтому

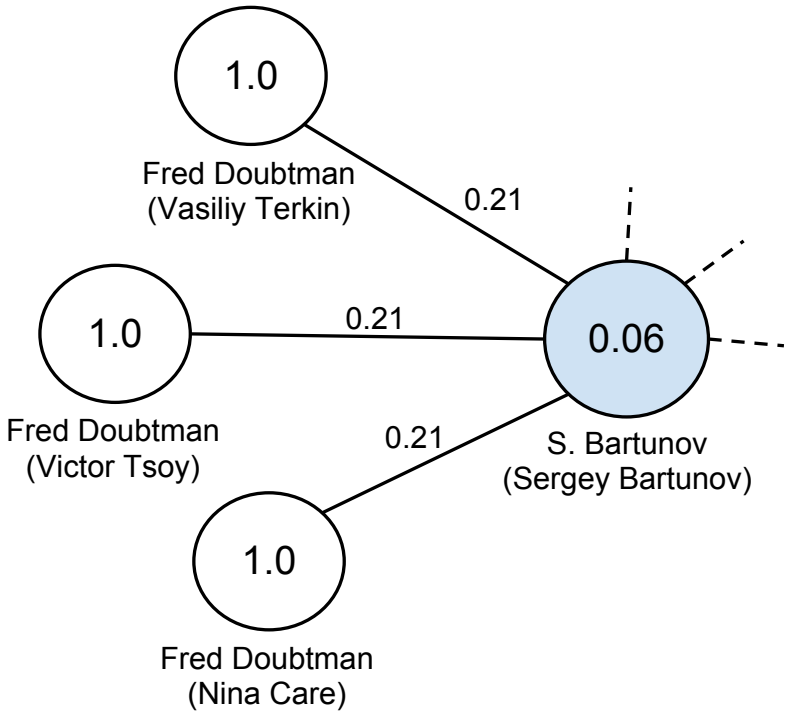
$$\Psi(\mathbf{y}_v, \mathbf{y}_u) = \begin{cases} \text{network-distance}(\mu(v), \mu(u)) & v \neq u \\ \infty & \text{иначе} \end{cases}$$

Эвристика

- Для смежных вершин выбирать проекции, которые дружат в графе B
-

$$\Psi(\mathbf{y}_v, \mathbf{y}_u) = \begin{cases} \text{network-distance}(\mu(v), \mu(u)) & v \neq u \text{ и } (\mu(v), \mu(u)) \in B \\ \infty & \text{иначе} \end{cases}$$

Очистка результатов



Плохие результаты

- Плохая связность (см. рисунок)
- Мало вершин с заранее известными проекциями
- ...

Вывод: результаты надо очищать

Очистка результатов - тривиальное решение

Взаимное проецирование

- Получить μ из A в B
- Получить ν из B в A
- Если $v \neq \nu(\mu(v))$, то $\mu(v) \leftarrow \mathbf{N}$

Свойства

- Просто и интуитивно
- Слишком грубая техника очистки результатов - не учитывает *причину* ошибки
- Требует ~ 2 раза больше времени

Очистка результатов - классификатор

Признаки

1. profile-distance$(v, \mathrm{pr}(v))$
2. Средняя графовая близость к проекциям смежных вершин
3. Доля заранее известных проекций среди смежных вершин
4. Взаимо-согласованность смежных вершин с заранее известными проекциями:

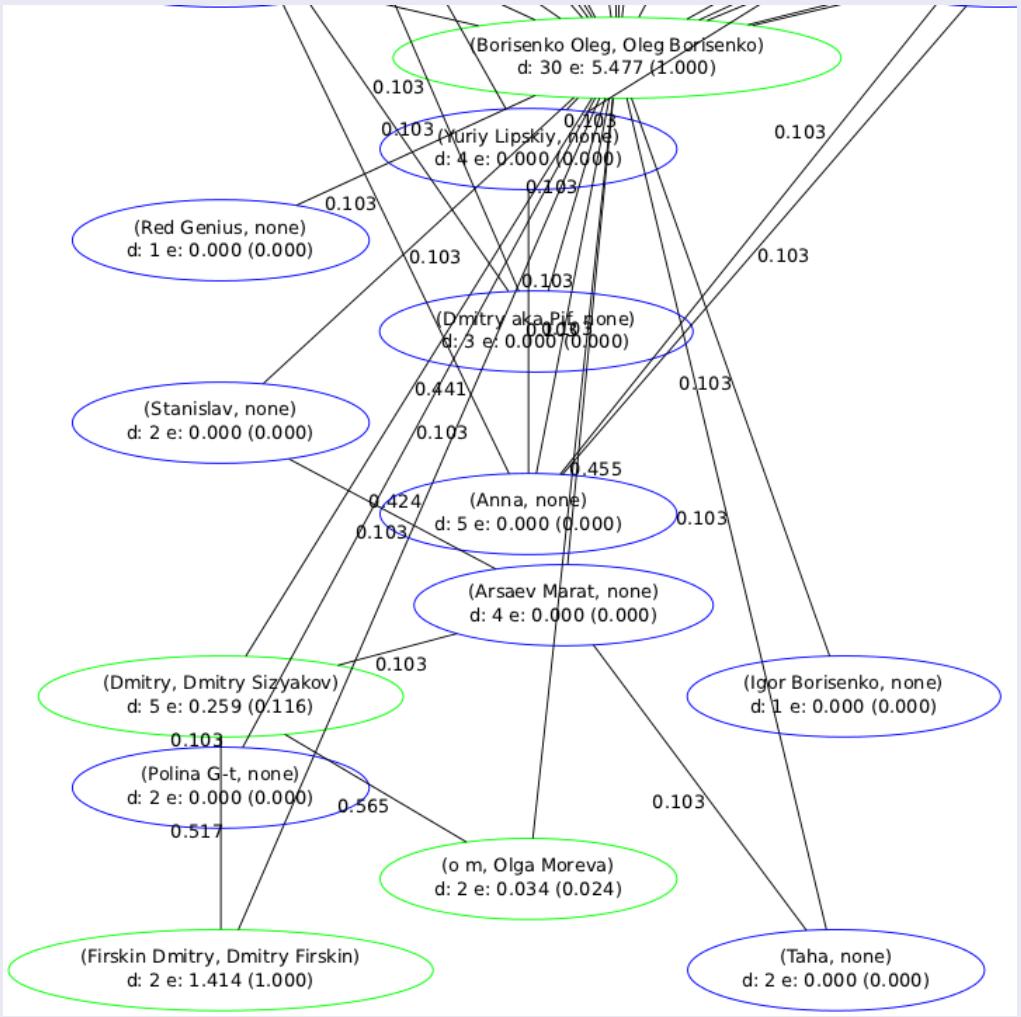
$$\frac{1}{n} \cdot \sum_{v} \frac{1}{n-1} \sum_{u \neq v} \text{network-distance}(\mathrm{pr}(v), \mathrm{pr}(u))$$

Сравнение классификаторов

алгоритм	полнота	точность	F_1
Naive Bayes	0.762	0.256	0.383
Support Vector Machine	0.662	0.935	0.775
C4.5	0.715	**0.939**	0.812
C4.5 с MultiBoosting	**0.844**	0.902	**0.872**

Результаты идентификации

Пример



Экспериментальные данные

Статистика

	Twitter	Facebook
Основная выборка		
# центральных пользователей	16	
# профилей	398	977
# связей	1 728	10 256
# сопоставленных профилей	141	
# заранее известных проекций	71	
Дополнительная выборка		
# центральных пользователей	17	
# профилей	1 499	7 425
# связей	15 943	172 219
# сопоставленных профилей	161	

Базовые алгоритмы

Максимальное парасочетание

- Каждому профилю из A нужно сопоставить не более одного профиля из B
- Сопоставленные профили должны быть как можно более похожи по полям профилей
- Значение функции похожести должно быть не ниже некоторого порога
- Порог максимизирует точность

Базовые алгоритмы

1. Взвешенная сумма значений вектора $V(v, \mu(v))$
2. $1 -$ profile-distance$(v, \mu(v))$

Оценка качества

Точность и полнота

$$\text{полнота} = \frac{\text{true-positives}}{\text{true-positives} + \text{false-negatives}}$$

$$\text{точность} = \frac{\text{true-positives}}{\text{true-positives} + \text{false-positives}}$$

- true-positive - профиль сопоставлен верно
- false-negative - профиль не был сопоставлен, а должен был быть
- false-positive - профиль сопоставлен неверно

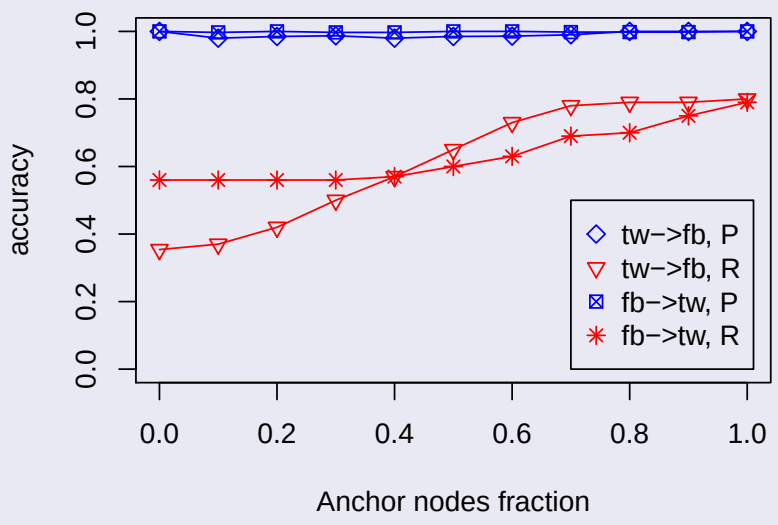
Эксперименты

Результаты на основной выборке

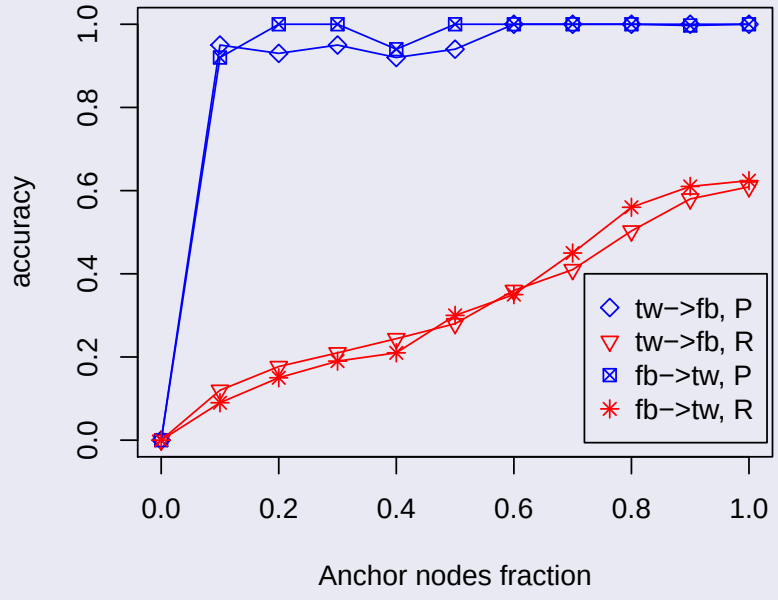
алгоритм	полн.	точн.	F_1
безразличные к направлению проекции			
Базовый 1 (взвешенная сумма)	0.45	0.94	0.61
Базовый 2 (вероятностная похожесть)	0.51	**1.0**	0.69
JLA, взаимн. проекц., аноним.	0.6	**1.0**	0.76
JLA, взаимн. проекц.	0.66	0.99	0.79
Twitter $\rightarrow$ Facebook			
JLA, анонимн. ($\Phi \equiv 0$)	0.62	**1.0**	0.77
JLA	0.79	**1.0**	0.89
Facebook $\rightarrow$ Twitter			
JLA, анонимн. ($\Phi \equiv 0$)	0.61	**1.0**	0.76
JLA	**0.8**	**1.0**	**0.89**

Эксперименты

Идентификация



Деанонимизация



Повторная идентификация

Мотивация

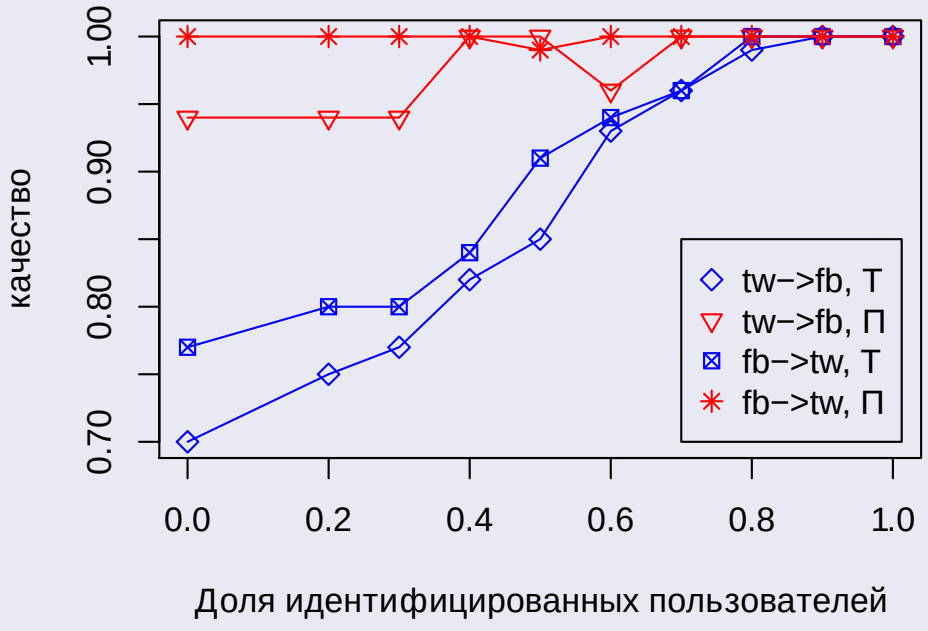
- Основная выборка мала
- Собрать хорошую выборку тяжело без помощи владельцев аккаунтов
- Нужно протестировать алгоритм автоматически на неразмеченной выборке

Постановка задачи

- При помощи второго базового алгоритма сопоставляются профили
- Фиксируется некоторая часть из них
- У всех возможных проекций для этой части профилей убирается вся информация, кроме связей
- Нужно найти проекции для этих профилей по связям

Повторная идентификация

Идентификация



Вывод

80% известных проекций достаточно чтобы определить оставшиеся 20%

Дальнейшее направление работы

- Собрать еще больше данных
  - Присылайте на sbartunov@gmail.com
- Не все связи одинаково полезны
- Ввести дополнительные бинарные факторы и подобрать к ним веса
- Корреляция между network-distance(v, u) и network-distance$(\mu(v), \mu(u))$