Министерство образования и науки Российской Федерации
«Московский физико-технический институт (государственный университет)»
Физтех-школа прикладной математики и информатики
Факультет управления и прикладной математики
Кафедра «Интеллектуальные системы»

Хайруллин Ринат Ильдарович

Автоматическое выделение именованных сущностей в коллекциях текстовых документов

03.04.01 – Прикладные математика и физика

Выпускная квалификационная работа
(Магистерская диссертация)

Научный руководитель:
д. ф.-м. н. Серебряков Владимир Алексеевич

Москва
2018

Аннотация

В работе рассматривается задача выделения именованных сущностей в коллекции документов, относящихся к одной тематике. Задача решается для случая с частичной разметкой выборки документов. Предложен алгоритм для анализа корпуса текстов, позвляющий выделить n-граммы, являющиеся именованными сущностями, с высоким значением полноты. Для выделенных n-грамм решается задача классификации. По корпусу документов строятся графы зависимостей, учитывающие совместное употребление выделенных n-грамм с глаголами и предлогами. На построенных графах задается функция потерь, для определения оптимальной разметки вершин графов классами именованных сущностей из некоторого заданного конечного множества. Проведен вычислительный эксперимент на корпусе разметченных документов конкурса FactRuEval и корпусе документов лаборатории LABINFORM.

Содержание

Введение

Задачу «Распознавание именованных сущностей» можно рассматривать как подзадачу задачи «Извлечение информации», где из неструктурированного корпуса документов необходимо извлечь структурированный текст. В задаче требуется выделить и классифицировать слова в документе на предопределенные классы. Термин «именованные» ограничивает задачу до поиска слов, которые человек может однозначно сопоставить с одним из классов. Первоначально на конференции MUC6 ставилась задача идентификации в документах имен людей, названий организаций и геолокаций. Впоследствии, с появлением задач обработки текстов, относящихся к узкоспециализированным предметным областям, таким как медицина и биология, стали выделять тематически-ориентированные классы именованных сущностей: названия генов и протеинов, названия лекарств, болезней и.т.д. Иными словами, понятие «именованная сущность» можно обобщенно определить, как лексикализованное[1] выражение, используемое для обозначения уникального объекта человеческих знаний о мире, а класс именованной сущности это семантическая категория (концепция высокого уровня), объединяющая объекты. Выделение именованных сущностей из коллекции документов, позволяет получить предварительное представление о содержании документов, помогает улучшить работу вопросно-ответных систем. Перечисленное делает задачу востребованной в рамках информационного поиска.

На текущий момент, выработано два основных подхода, которые используются для решения задач распознавания именованных сущностей: составленные экспертами правила вывода и методы машинного обучения.

Обзор существующих методов

Правила вывода представляют собой лексико-синтаксические шаблоны, созданные с участием ученых-лингвистов, программистов и математиков. Методы основанные на пра-

[1] *прим. из Википедии. Лексикализация является механизмом пополнения как общеупотребительной, так и специальной и терминологической лексики, в результате которой лексикализованные сочетания в конечном итоге входят в языковую норму*

вилах вывода точны, не требуют обучающей выборки, но обладают рядом существенных недостатков:

1. зависимость от структуры языка,
2. плохая обобщающая способность и расширяемость другие тематики,
3. сложность в разработке — требуется непосредственное взаимодействие различных групп ученых,
4. эффективность работы правил зависит от тематики документов, так как требуется достаточный формализм в описании именованной сущности.

Простые правила вывода могут задаваться регулярными выражениями, которые выделяют последовательности слов, соответствующие простым шаблонам. Для русского языка активно разрабатывается библиотека *Natasha*[2], в ее основе лежат правила вывода, которые достаточно хорошо умеют классифицировать имена персон, даты, суммы денег.

Большей популярностью, пользуются алгоритмы машинного обучения с учителем. В задаче распознавания именованных сущностей требуется построить модель для предсказания меток классов $\mathbf{y} = \{y_1, \ldots, y_T\}$, $y_i \in Y$ для произвольных предложений $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$. Алгоритмы настраиваются по обучающей выборке $\{(\mathbf{x}, \mathbf{y})^k\}_{k=1}^N$, где $\mathbf{x}_i$ – признаковое описание i-го слова. Каждый $\mathbf{x}_i$ – содержит различную информацию о слове на позиции i, например:

- о приставках и суффиксах в слове,
- о стилистике в написании слова,
- о части речи, склонениях и.т.д.

При распознавании **Персон**, **Организаций**, **Геолокаций** обычно выделяют следущие метки классов:

$$Y = \{\text{B-Per}, \text{I-Per}, \text{B-Loc}, \text{I-Loc}, \text{B-Org}, \text{I-Org}, \text{B-Misc}, \text{I-Misc}, \text{O}\}.$$

Одними из первых предложенных алгоритмов для решения задачи были скрытые марковские цепи (HMM) [1]. В основе методов лежит принцип максимизации правдоподобия совметсной функции распределения $p(\mathbf{y}, \mathbf{x})$, требуется задать функцию распределения на множестве всевозможных комбинаций наблюдаемых $\mathbf{x}$ и классов $\mathbf{y}$. На практике, для решения задачи не требуется учитывать всевозможные взаимосвязи в данных, более того,

[2] https://github.com/natasha/natasha

в худшем случае, для настройки параметров модели может потребоваться перебор всевозможных последовательностей наблюдаемых и скрытых состояний, что для больших размеров входных данных может привести к неразрешимости задачи [2].

Другое семейство методов, основано на дискриминативных моделях: Maximum Entropy Models (MEM) и Conditional Random Fields (CRF) [3, 4]. Задача распознавания именованных сущностей для этих моделей сводится к вычислению маргинальных распрелелений $p(\mathbf{y}|\mathbf{x})$ и прогнозированию с помощью них классов слов в предложении $\hat{\mathbf{y}} = \operatorname{argmax}_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$. Несмотря на схожесть моделей MEM и CRF, модель CRF обладает существенным преимуществом, она лишена проблемы *смещения метки* (label bias). Проблема *смещения метки* заключается в том, что *будущие* наблюдения не влияют на распределение более ранних состояний [2]. К общими недостатками MEM и CRF можно отнести то, что для обучения этих моделей требуется большой объем аннотированных документов и тщательный отбор признаков. Проблему выделения признаков, решают алгоритмы автоматической генерации векторных представлений слов (*word embedings*) [5–8]. В работе [9] предложен гибридный Bi-LSTM-CRF алгоритм, для распознавания Персон, Организаций и Геолокаций в документах на русском языке, в качестве признаков используются векторные представления слов из библиотеки *FastText*[3] и Bi-LSTM нейросеть для получения векторов на основе символов.

В условиях малого объема обучающей выборки для распознавания именованных сущностей применяются алгоритмы частичного обучения (semi-supervised learning). В работе [10] предложен достаточно общий подход к решению задачи:

1. обучить CRF модель на размеченной выборке,
2. применить модель к неразмеченной выборке,
3. расширить размеченную выборку, новыми объектами из п.2,
4. повторять шаги 1-3 пока алгоритм не обучится.

В работе [11] предлагается сгенерировать векторные представления слов на основе статистик их совстречаемости в большом корпусе текстов, а затем использовать получившиеся векторы в качестве расширения признаков при построении CRF/Semi-Supervised-CRF модели.

[3] https://fasttext.cc/

Проблемы существующих методов

Проблемы использования CRF подходов: модели необходимо периодически переобучать, из-за устаревания словаря, в особенности, это касается задач, которые относятся к узкоспециализированным темам, таким как биология или медицина, выделение признаков для описания слов является трудоемкой задачей, в зависимости от того, какие классы именованных сущностей требуется распознать, признаки могут изменяться [2, 3, 12]. Проблемы разрешения лексической многозначности и распознавания редких слов присущи, в целом, задаче распознавнаия именованных сущностей. Применение описанных выше алгоритмов для русского языка, осложнаяется еще и тем, что доступно малое количество качественно размеченных корпусов документов.

Цель работы. Целью работы является разработка алгоритма классификации лексически многозначных именованных сущностей при наличии частичной разметки корпуса документов.

Метод исследования. Для достижения поставленной цели предлагается:

- ввести эврестические предположения о свойствах n-грамм - именованных сущностей, построить алгоритм выделения таких n-грамм в корпусе текстов,
- моделировать классы отдельно для каждого вхождения именованной сущности в корпус документов,
- зная метки классов небольшой части n-грамм в корпусе, построить алгоритм получения меток классов на оставшейся части выборки.

Работа организована следующим образом. Сперва опишем постановку задачи и алгоритм классификации. Затем опишем способ применения алгоритма к задаче распознавания Персон, Организаций, Геолокаций. Протестируем полученную модель на размеченном корпусе текстов на русском языке.

Постановка задачи

Введем некоторые определения.

Определение 1. N-граммой будем называть некоторую непрерывную подпоследовательность слов $\{w_{s,k_1}, \ldots, w_{s,k_t}\}$ в предложении.

Определение 2. Именованной сущностью ϕ будем называть n-грамму которая удовлетворяет следующим свойствам:

1. последовательность слов n-граммы, сформирована неслучайным образом,
2. n-грамма является полной по включению последовательностью слов,
3. последовательность частей речи n-граммы соответствует виду ([причастие]{0, 1}[прилагательное]{0, 2}[существительное]+).

Определение 3. Связью r будем называть n-грамму, последовательность частей речи которой удовлетворяет регулярному выражению
$([предлог]|[глагол] + [предлог]\{0, 1\})$.

Будем считать, что все определенные графы имеют конечное число вершин.

Определение 4. Неориентированный граф $\mathcal{G} = (W, E)$, называется двудольным, если его множество вершин можно разбить на две части $W = U \sqcup V$ так, что:

- ни одна из вершин в U не соединена с вершинами в U,
- ни одна из вершин в V не соединена с вершинами в V.

Определение 5. Матрица смежности графа $\mathcal{G}$ с конечным числом вершин n – это квадратная матрица $\boldsymbol{\mathcal{G}}$ размера n, $g_{i,j}$ – элемент которой равен единице, если между i и j вершиной графа существует ребро.

Пусть даны корпус текстов $\mathcal{D} = \{d_1, \ldots, d_D\}$, относящийся к некоторой теме, и некоторое конечное множество классов $Y = \{y_1, \ldots, y_T\}$. Документ $d \in \mathcal{D}$ представляется в виде упорядоченной последовательности предлжоений $d = \{s_1, \ldots, s_d\}$. Каждое предложение s, в свою очередь, представляется ввиде упорядоченной последовательности слов

$s = \{w_1, \ldots, w_s\}$, $w \in \mathcal{V}$, где $\mathcal{V}$ – множество всех слов без повторений в документах коллекции $\mathcal{D}$.

В первую очередь требуется выделить множество всех уникальных именованных сущностей $\mathcal{Q}$ в корпусе документов. Чтобы избавиться от привязки к тематике корпуса документов, необходимо построить алгоритм выделения именованных сущностей, исходя из общих лингвистических предположений. Для решения проблемы с лексической многозначностью, будем моделировать класс $y \in Y$ отдельно для каждой позиции именованной сущности в корпусе документов. Предлагается моделировать класс y на основе словесного представления именованной сущности и ее контекстной совстречаемости с какой-либо из связей r в предложениях. Из-за того, что именованная сущность может редко встречаться в контексте какой-либо связи r, необходимо в процессе обучения производить "мягкую" кластеризацию на множестве всех уникальных связей $\mathcal{R}p$ и при определении класса также учитывать кластерную принадлежность связи. При таком подходе к решению, задачу распознавания именованных сущностей можно сформулировать, как задачу трансдуктивного обучения на графовом представлении корпуса текстов.

Рассматриваемый алгоритм

Алгоритм трансдуктивного обучения на графе

Опишем общую идею и постановку задачи оптимизации предложенную в работе [13]. Пусть по корпусу документов $\mathcal{D}$ построены: множество всех уникальных именованных сущностей $\mathcal{Q}$, множество всех их словопозиций $\mathcal{M}$ и множество всех уникальных связей $\mathcal{R}p$. Пусть также известны метки классов на некотором подмножестве $\mathcal{M}_0 \subset \mathcal{M}$. Построим графы на вышеперечисленных множествах:

$$\boldsymbol{\mathcal{G}}_{\mathcal{Q}} = (\mathcal{M} \sqcup \mathcal{Q}, \mathcal{E}_{\mathcal{Q}}) \quad \boldsymbol{\mathcal{G}}_L = (\mathcal{M} \sqcup \mathcal{R}p, \mathcal{E}_{left}) \quad \boldsymbol{\mathcal{G}}_R = (\mathcal{M} \sqcup \mathcal{R}p, \mathcal{E}_{right})$$
$$\boldsymbol{\mathcal{W}}_L = (\mathcal{Q} \sqcup \mathcal{R}p, \mathcal{E}_L, \nu) \quad \boldsymbol{\mathcal{W}}_R = (\mathcal{Q} \sqcup \mathcal{R}p, \mathcal{E}_R, \nu) \quad \boldsymbol{\mathcal{W}}_{\mathcal{M}} = (\mathcal{M}, \mathcal{E}_{\mathcal{M}}, f)$$
$$\nu : \mathcal{E}_{\{L,R\}} \to \mathbb{R}_+, \quad f : \mathcal{E}_{\mathcal{M}} \to \mathbb{R}_+$$

$\boldsymbol{\mathcal{G}}_{\mathcal{Q}}, \boldsymbol{\mathcal{G}}_L, \boldsymbol{\mathcal{G}}_R, \boldsymbol{\mathcal{W}}_L, \boldsymbol{\mathcal{W}}_R$ – двудольные графы, $\boldsymbol{\mathcal{W}}_{\mathcal{M}}$ – граф зависимостей между элементами множества $\mathcal{M}$. f–некоторая функция близости заданная на множестве ребер графа $\boldsymbol{\mathcal{W}}_{\mathcal{M}}$. ν – функция, которая каждому ребру в графе сопоставляет частоту, с которой именованная сущность $q \in \mathcal{Q}$ и связь $r \in \mathcal{R}p$ встречаются в корпусе $\mathcal{D}$. Индикаторы $\{left, right\}$ указывают на положение $m \in \mathcal{M}$ и $r \in \mathcal{R}p$ друг относительно друга в предложении. Будем считать, что графы представлены в виде матриц смежности. Тогда $\boldsymbol{\mathcal{W}}_{\{\mathcal{L},\mathcal{R}\}}$ можно задать через произведение матриц

$$\boldsymbol{\mathcal{W}}_{\{L,R\}} = \boldsymbol{\mathcal{G}}_{\mathcal{Q}}^T \boldsymbol{\mathcal{G}}_{\{L,R\}}.$$

Задача определить класс $y \in Y$ для всех вершин $m \in \mathcal{M} \setminus \mathcal{M}_0$. Введем индикаторные матрицы на множествах вершин:

$$\boldsymbol{\mathcal{Y}} \in \{0,1\}^{|\mathcal{M}| \times T}, \quad \boldsymbol{\mathcal{Y}}_0 \in \{0,1\}^{|\mathcal{M}| \times T}, \quad \boldsymbol{\mathcal{C}} \in \mathbb{R}^{|\mathcal{Q}| \times T} \quad \boldsymbol{\mathcal{P}}_{\mathcal{L}} \in \mathbb{R}^{|\mathcal{R}p| \times T}, \quad \boldsymbol{\mathcal{P}}_{\mathcal{R}} \in \mathbb{R}^{|\mathcal{R}p| \times T}$$

Элемент (i, j) индикаторных матриц, определяет насколько сильно вершина i относится к классу j. В матрице $\boldsymbol{\mathcal{Y}}_0$ на позициях элементов $m \in \mathcal{M}_0$ стоят единицы для соответств-

тующих классов, а на всех остальных нули. Зададим функцию потерь:

$$\begin{aligned}\Omega_{\gamma,\mu}\left(\boldsymbol{\mathcal{Y}},\boldsymbol{\mathcal{C}},\boldsymbol{\mathcal{P}}_L,\boldsymbol{\mathcal{P}}_R\right) = \|\boldsymbol{\mathcal{Y}} - f\left(\boldsymbol{\mathcal{G}}_Q\boldsymbol{\mathcal{C}},\boldsymbol{\mathcal{G}}_L\boldsymbol{\mathcal{P}}_L,\boldsymbol{\mathcal{G}}_R\boldsymbol{\mathcal{P}}_R\right)\|_F^2 + \mu\|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{Y}}_0\|_F^2 \\ + \frac{\gamma}{2}\sum_{i,j}^{|\mathcal{M}|} W_{\mathcal{M},i,j}\|\frac{\boldsymbol{\mathcal{Y}}_i}{\sqrt{\mathbf{D}_{\mathcal{M},i,i}}} - \frac{\boldsymbol{\mathcal{Y}}_j}{\sqrt{\mathbf{D}_{\mathcal{M},j,j}}}\|_2^2 \\ + \sum_{Z\in\{L,R\}}\sum_{i}^{|\mathcal{Q}|}\sum_{j}^{|\mathcal{R}p|} W_{Z,i,j}\|\frac{\boldsymbol{\mathcal{C}}_i}{\sqrt{\mathbf{D}^{\mathcal{Q}}_{Z,i,i}}} - \frac{\boldsymbol{\mathcal{P}}_{Z,j}}{\sqrt{\mathbf{D}^{\mathcal{R}p}_{Z,i,i}}}\|_2^2\end{aligned} \tag{1}$$

$$\mathbf{D}_{\mathcal{M},i,i} = \sum_{j}^{|\mathcal{M}|} W_{\mathcal{M},i,j}, \quad \mathbf{D}^{\mathcal{Q}}_{Z,i,i} = \sum_{j}^{|\mathcal{R}p|} W_{Z,i,j}, \quad \mathbf{D}^{\mathcal{R}p}_{Z,j,j} = \sum_{i}^{|\mathcal{Q}|} W_{Z,i,j}$$

Первое слагамое функции потерь моделирует метки классов на множестве вершин $\mathcal{M}$ в виде некоторой функциональной зависимости $f(\cdot)$ от меток классов на множествах $\mathcal{Q}$ и $\mathcal{R}p$. В работе для исследований выберем функцию $f(x,y,z) = x+y+z$. Второе слагаемое штрафует функцию если метки классов на размеченной части выборки сильно не совпадают с моделируемыми. Последние два слагаемых функции потерь (1) вносят штраф, если метки классов на вершинах связанных ребром с большим весом отличаются. Нормализуем матрицы $\boldsymbol{\mathcal{W}}_{\mathcal{M}}, \boldsymbol{\mathcal{W}}_{\{L,R\}}$

$$\boldsymbol{\mathcal{S}}_{\mathcal{M}} = \mathbf{D}_{\mathcal{M}}^{-\frac{1}{2}}\boldsymbol{\mathcal{W}}_{\mathcal{M}}\mathbf{D}_{\mathcal{M}}^{-\frac{1}{2}}, \quad \boldsymbol{\mathcal{S}}_{\{L,R\}} = {\mathbf{D}^{\mathcal{Q}}_{\{L,R\}}}^{-\frac{1}{2}}\boldsymbol{\mathcal{W}}_{\{L,R\}}{\mathbf{D}^{\mathcal{R}p}_{\{L,R\}}}^{-\frac{1}{2}},$$

и введем матрицу Лапласа [14]

$$\boldsymbol{\mathcal{L}}_{\mathcal{M}} = \mathbf{I} - \boldsymbol{\mathcal{S}}_{\mathcal{M}}.$$

Тогда функцию потерь (1) можно переписать в следующем виде

$$\begin{aligned}\Omega_{\gamma,\mu}\left(\boldsymbol{\mathcal{Y}},\boldsymbol{\mathcal{C}},\boldsymbol{\mathcal{P}}_L,\boldsymbol{\mathcal{P}}_R\right) = \|\boldsymbol{\mathcal{Y}} - f\left(\boldsymbol{\mathcal{G}}_Q\boldsymbol{\mathcal{C}},\boldsymbol{\mathcal{G}}_L\boldsymbol{\mathcal{P}}_L,\boldsymbol{\mathcal{G}}_R\boldsymbol{\mathcal{P}}_R\right)\|_F^2 + \mu\|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{Y}}_0\|_F^2 \\ + \gamma Tr\left(\boldsymbol{\mathcal{Y}}^T\boldsymbol{\mathcal{L}}_{\mathcal{M}}\boldsymbol{\mathcal{Y}}\right) \\ + \sum_{Z\in\{L,R\}} Tr\left(\boldsymbol{\mathcal{C}}^T\boldsymbol{\mathcal{C}} + \boldsymbol{\mathcal{P}}_Z^T\boldsymbol{\mathcal{P}}_Z - 2\boldsymbol{\mathcal{C}}^T\boldsymbol{\mathcal{S}}_Z\boldsymbol{\mathcal{P}}_Z\right),\end{aligned} \tag{2}$$

$Tr\left(\cdot\right)$ – след матрицы, γ и μ – положительные гиперпараметры, учитывающие влияние штрафов на функцию потерь.

Задача неотрицательного матричного разложения

Начнем с описания задачи неотрицательного матричного разложения. Дана матрица $\mathbf{F} \in \mathbb{R}_{\geq 0}^{m\times n}$, где m объектов описаны n признаками. Требуется найти разложение матрицы:

$$\mathbf{F} \approx \mathbf{U}\mathbf{V}^T, \quad \mathbf{U} \in \mathbb{R}_{\geq 0}^{m\times k}, \quad \mathbf{V} \in \mathbb{R}_{\geq 0}^{n\times k}$$

, где $\mathbf{U}$ – представление исходной матрицы в некотором пространстве меньшей размерности $k < \min(m, n)$, $\mathbf{V}$ – базис. В работе [15] для постановки задачи в виде

$$\min_{\mathbf{U},\mathbf{V}} \quad \|\mathbf{F} - \mathbf{U}\mathbf{V}^T\|_F^2$$
$$\text{s.t. } \mathbf{U} \geq 0, \quad \mathbf{V} \geq 0$$

предложили следующие правила обновления матриц $\mathbf{U}, \mathbf{V}$

$$U_{i,k} \leftarrow U_{i,k} \frac{(\mathbf{F}\mathbf{V})_{i,k}}{(\mathbf{U}\mathbf{V}^T\mathbf{V})_{i,k}}$$
$$V_{j,k} \leftarrow V_{j,k} \frac{(\mathbf{F}^T\mathbf{U})_{j,k}}{(\mathbf{V}\mathbf{U}^T\mathbf{U})_{j,k}}$$

и доказали, что итеративный процесс оптимизации с помощтю этих правил приводит к невозрастанию функции потерь, при этом сохраняя неотрицательность матриц. В работе [16] показано, что вообще говоря, итерационный процесс не обязательно сходится к локальному минимуму. Также алгоритм чувствителен к выбору начального приближения матриц $\mathbf{U}, \mathbf{V}$.

Задача неотрицательного матричного разложения по совокупности компонент $\mathbf{U}, \mathbf{V}$ является невыпуклой, для большинства задач этого класса удается показать только сходимость к стационарной точке. Для многих мультипликативных алгоритмов, как альтернативу сходимости, можно использовать тот факт, что на каждой итерации алгоритма функция потерь не возрастает, соответственно это служит объяснением применимости алгоритма [17].

Эквивалентность NMF и PLSI

Даны m документов с размером словаря n, матрица $\mathbf{F} \in \mathbb{R}_{\geq 0}^{m \times n}$ задает частоты слов в документах, $F_{i,j}$ – частота слова w_i в документе d_j. Если матрицу $\mathbf{F}$ нормировать на величину $\sum_{i,j} F_{i,j}$, то получим матрицу совместного распределения слов в документах $p(w_i, d_j) = F_{i,j}$. В работах [18, 19] показывается, что если матрица $\mathbf{F}$ – стохастическая, то задача PLSI эквивалентна задаче NMF с функцией потерь в виде обобщенной дивергенции Кульбака-Лейблера. Тогда матрицы $\mathbf{U}, \mathbf{V}$ полученные в результате оптимизации NMF

задачи имеют формальную вероятностную интерпретацию

$$\mathbf{U} \in \mathbb{R}^{m\times k}_{\geq 0}, \quad \mathbf{V} \in \mathbb{R}^{n\times k}_{\geq 0}, \quad \mathbf{Q} \in \mathbb{R}^{k\times k}_{\geq 0}, \quad Q_{k,k} = \sum_i V_{i,k}$$

$$\|\mathbf{F}\|_{\ell_1} = 1, \quad \mathbf{U}\mathbf{V}^T = (\mathbf{U}\mathbf{Q})\left(\mathbf{Q}^{-1}\mathbf{V}^T\right)$$

$$[p\,(w \mid k)]^T = \left(\mathbf{Q}^{-1}\mathbf{V}^T\right), \quad [p\,(d,k)] = (\mathbf{U}\mathbf{Q}), \quad \sum_{d,k} p(d,k) = 1, \quad \sum_w p\,(w \mid k) = 1$$

$$1 = \|\mathbf{F}\|_{\ell_1} \approx \sum_k \|\sum_i U_{i,k} \sum_j V_{j,k}\|_{\ell_1} = \sum_k \|\sum_i U_{i,k}\|_{\ell_1}$$

Многофакторное матричное разложение

Опишем алгоритм предложенный в работе [20]. Пусть есть m объектов, и задано v различных признаковых представлений для этих объектов. Иными словами, дано множество матриц объект признак $\{\mathbf{F}^1, \ldots, \mathbf{F}^v\}$. В каждой из матриц $\mathbf{F}^e$ одинаковое число объектов m и различное число признаков n_e. Требуется для каждого из матриц $\mathbf{F}^e$ найти низкоранговое представление $\mathbf{U}^e$

$$\mathbf{F}^e \approx \mathbf{U}^e(\mathbf{V}^e)^T, \quad \mathbf{F}^e \in \mathbb{R}^{m\times n_e}_{\geq 0}, \quad \mathbf{U}^e \in \mathbb{R}^{m\times k}_{\geq 0}, \quad \mathbf{V}^e \in \mathbb{R}^{n_e\times k}_{\geq 0},$$

требуется также учесть, что новые представления объектов $\mathbf{U}^e$ должны быть согласованы между собой. Однако, они могут быть не сопоставимы в одном маштабе. Чтобы избавится от несогласованности в различных представлениях данных, авторы используют идею описанную в предыдущей главе. Если нормализовать исходные матрицы $\mathbf{F}^e$ по ℓ_1 норме и наложить ограничение $\|\mathbf{V}^e_{:,k}\|_{\ell_1} = 1$, то $\|\mathbf{U}^e\|_{\ell_1} = 1$, тогда различные представления можно будет сравнивать между собой. Постановка задачи оптимизации

$$\min_{\{\mathbf{U}^e,\mathbf{V}^e\},\mathbf{U}^*} \quad \sum_{e=1}^{v}\left(\|\mathbf{F}^e - \mathbf{U}^e(\mathbf{V}^e)^T\|_F^2 + \alpha_e\|\mathbf{U}^e - \mathbf{U}^*\|_F^2\right)$$

$$\text{s.t. } \{\mathbf{U}^e, \mathbf{V}^e\}, \mathbf{U}^* \geq 0, \quad \forall 1 \leq t \leq k \quad \|\mathbf{V}^e_{:,k}\|_{\ell_1} = 1,$$

где $\mathbf{U}^*$ – матрица консенсуса, которая добавляет штраф к функции потерь, если различные $\mathbf{U}^e$ сильно различны между собой. Избавится от ограничений $\|\mathbf{V}^e_{:,k}\|_{\ell_1} = 1$ можно путем введения матриц

$$\mathbf{Q}^e = Diag\left(\sum_{i=1}^{n_e} V^e_{i,1}, \ldots, \sum_{i=1}^{n_e} V^e_{i,k}\right).$$

Тогда задача оптимизации примет следующий вид

$$\min_{\{\mathbf{U}^e,\mathbf{V}^e\},\mathbf{U}^*} \quad \underbrace{\sum_{e=1}^{v}\left(\overbrace{\|\mathbf{F}^e - \mathbf{U}^e(\mathbf{V}^e)^T\|_F^2 + \alpha_e\|\mathbf{U}^e\mathbf{Q}^e - \mathbf{U}^*\|_F^2}^{\delta}\right)}_{\mathcal{J}} \tag{3}$$

$$\text{s.t. } \{\mathbf{U}^e, \mathbf{V}^e\}, \mathbf{U}^* \geq 0.$$

Приведем правила обновления матриц [20].

$$V_{i,k} \leftarrow V_{i,k} \frac{\left(\mathbf{F}^T\mathbf{U}\right)_{i,k} + \alpha \sum_{j=1}^{m} U_{j,k} U^*_{j,k}}{\left(\mathbf{V}\mathbf{U}^T\mathbf{U}\right)_{i,k} + \alpha \left(\sum_{t=1}^{m} U^2_{t,k}\right) \left(\sum_{j=1}^{n_e} V_{j,k}\right)} \tag{4}$$

$$U_{j,k} \leftarrow U_{j,k} \frac{\left(\mathbf{F}\mathbf{V} + \alpha\mathbf{U}^*\right)_{j,k}}{\left(\mathbf{U}\mathbf{V}^T\mathbf{V} + \alpha\mathbf{U}\right)_{j,k}} \tag{5}$$

$$\mathbf{U}^* = \frac{\sum_{e=1}^{v} \mathbf{U}^e \mathbf{Q}^e}{\sum_{e=1}^{v} \alpha_e} \tag{6}$$

Algorithm 1 Multi-view NMF

Вход: $\{\mathbf{F}^1, \ldots, \mathbf{F}^v\}$, $\{\alpha_1, \ldots, \alpha_v\}$, число кластеров k

Выход: $\{\mathbf{U}^1, \ldots, \mathbf{U}^v\}$, $\{\mathbf{V}^1, \ldots, \mathbf{V}^v\}$, $\mathbf{U}^*$

нормализуем $\{\mathbf{F}^1, \ldots, \mathbf{F}^v\}$ по ℓ_1 норме
задаем начальное приближение для $\{\mathbf{U}^1, \ldots, \mathbf{U}^v\}$, $\{\mathbf{V}^1, \ldots, \mathbf{V}^v\}$, $\mathbf{U}^*$
повторять
  для $e = 1, \ldots, v$
    повторять
      обновляем $\mathbf{V}$ с помощью (4)
      $\mathbf{U} \leftarrow \mathbf{U}\mathbf{Q}$, $\mathbf{V} \leftarrow \mathbf{V}\mathbf{Q}^{-1}$
      $\mathbf{U}$ с помощью (5)
    пока δ не сойдется
  обновляем $\mathbf{U}^*$ с помощью (6)
пока $\mathcal{J}$ не сойдется

Алгоритм ClusType

Пусть всех связей $r \in \mathcal{R}p$ даны матрицы признаков:

- $\mathbf{F}_{context}$ – векторы контекстных признаков,
- $\mathbf{F}_{character}$ – векторы символьных признаков.

Таким образом каждый объект $r \in \mathcal{R}p$ описывается с помощью четырех признаковых представлений

$$\mathfrak{F} = \{\boldsymbol{\mathcal{P}}_L, \boldsymbol{\mathcal{P}}_R, \mathbf{F}_{context}, \mathbf{F}_{character}\}.$$

Совместная функция потерь для классификации вершин графа с кластеризацией связей

$$\mathcal{O}_{\gamma,\mu,\alpha} = \Omega_{\gamma,\mu}\left(\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{P}}_L, \boldsymbol{\mathcal{P}}_R\right) + \underbrace{\sum_{e=1}^{|\mathfrak{F}|} \left(\overbrace{\|\mathbf{F}^e - \mathbf{U}^e(\mathbf{V}^e)^T\|_F^2 + \alpha\|\mathbf{U}^e\mathbf{Q}^e - \mathbf{U}^*\|_F^2}^{\delta_e} \right)}_{\mathcal{J}_\alpha} \tag{7}$$

Задача оптимизации

$$\begin{aligned} &\min_{\boldsymbol{\mathcal{Y}},\boldsymbol{\mathcal{C}},\boldsymbol{\mathcal{P}}_{\{L,R\}},\{\mathbf{U}^e,\mathbf{V}^e,\beta_e\},\mathbf{U}^*} \quad \mathcal{O}_{\gamma,\mu,\alpha} \\ &\text{s.t.}\{\mathbf{U}^e,\mathbf{V}^e\},\mathbf{U}^* \geq 0, \sum_e \exp(-\beta_e) = 1, \boldsymbol{\mathcal{Y}} \in \{0,1\}^{|\mathcal{M}|\times T}, \boldsymbol{\mathcal{Y}}\mathbf{1} = \mathbf{1} \end{aligned} \tag{8}$$

Так как размерность T может быть меньше числа кластеров при работе алгоритма, к оптимизационной задаче добавлено ограничение на β_e, чтобы избежать тривиальных решений при матричном разложении. Последние два ограничения в задаче (8) приводят к NP-полноте. Поэтому будет решаться задача вещественнозначной релаксации, при отсутствии этих ограничений.

Класс для вершины $m \in \mathcal{M}$ будет определяться как

$$y(m) = \underset{t}{\operatorname{argmax}}\, \boldsymbol{\mathcal{Y}}_m.$$

Правила обновления матриц $\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{P}}_L, \boldsymbol{\mathcal{P}}_R$ [13]

$$\boldsymbol{\mathcal{Y}} = \left[(1+\mu)\boldsymbol{\mathcal{I}} + \gamma\boldsymbol{\mathcal{L}}_{\mathcal{M}}\right]^{-1} \left(f\left(\boldsymbol{\mathcal{G}}_Q\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{G}}_L\boldsymbol{\mathcal{P}}_L, \boldsymbol{\mathcal{G}}_R\boldsymbol{\mathcal{P}}_R\right) + \mu\boldsymbol{\mathcal{Y}}_0\right) \tag{9}$$

$$\boldsymbol{\mathcal{C}} = \frac{1}{2}\left[\boldsymbol{\mathcal{S}}_L\boldsymbol{\mathcal{P}}_L + \boldsymbol{\mathcal{S}}_R\boldsymbol{\mathcal{P}}_R + \boldsymbol{\mathcal{G}}_{\mathcal{Q}}^T\left(\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{G}}_L\boldsymbol{\mathcal{P}}_L - \boldsymbol{\mathcal{G}}_R\boldsymbol{\mathcal{P}}_R\right)\right] \tag{10}$$

$$\boldsymbol{\mathcal{P}}_L = \left[(1+\beta_L)\boldsymbol{\mathcal{I}} + \boldsymbol{\mathcal{G}}_L^T\boldsymbol{\mathcal{G}}_L\right]^{-1}\left[\boldsymbol{\mathcal{S}}_L^T\boldsymbol{\mathcal{C}} + \boldsymbol{\mathcal{G}}_L^T\left(\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{G}}_{\mathcal{Q}} - \boldsymbol{\mathcal{G}}_R\boldsymbol{\mathcal{P}}_R\right) + \beta_L\mathbf{U}_L\mathbf{V}_L\right] \tag{11}$$

$$\boldsymbol{\mathcal{P}}_R = \left[(1+\beta_R)\boldsymbol{\mathcal{I}} + \boldsymbol{\mathcal{G}}_R^T\boldsymbol{\mathcal{G}}_R\right]^{-1}\left[\boldsymbol{\mathcal{S}}_R^T\boldsymbol{\mathcal{C}} + \boldsymbol{\mathcal{G}}_R^T\left(\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{G}}_{\mathcal{Q}} - \boldsymbol{\mathcal{G}}_L\boldsymbol{\mathcal{P}}_L\right) + \beta_L\mathbf{U}_R\mathbf{V}_R\right] \tag{12}$$

Вообще говоря, правила (9)-(12) не гарантируют сохранение неотрицательности матриц $\boldsymbol{\mathcal{P}}_L, \boldsymbol{\mathcal{P}}_R$. По этому требуется модификация правил (4) - (6), так, чтобы после итераций обновления матрицы оставались неотрицательными [21].

Декомпозиция матриц

$$\mathbf{F} = \mathbf{F}_+ - \mathbf{F}_-, \quad F_{+,i,j} = \frac{|F_{i,j}| + F_{i,j}}{2}, \quad F_{-,i,j} = \frac{|F_{i,j}| - F_{i,j}}{2}$$

Измененные правила обновления для Multi-view NMF

$$V_{i,k} \leftarrow V_{i,k}\frac{\left(\mathbf{F}_+^T\mathbf{U}\right)_{i,k} + \alpha\sum_{j=1}^m U_{j,k}U_{j,k}^*}{\left(\mathbf{V}\mathbf{U}^T\mathbf{U}\right) + \mathbf{F}_-^T\mathbf{U}_{i,k} + \alpha\left(\sum_{t=1}^m U_{t,k}^2\right)\left(\sum_{j=1}^{n_e} V_{j,k}\right)} \tag{13}$$

$$U_{j,k} \leftarrow U_{j,k}\frac{\left(\mathbf{F}_+\mathbf{V} + \alpha\mathbf{U}^*\right)_{j,k}}{\left(\mathbf{U}\mathbf{V}^T\mathbf{V} + \mathbf{F}_-\mathbf{V} + \alpha\mathbf{U}\right)_{j,k}} \tag{14}$$

$$\mathbf{U}^* = \frac{\sum_{e=1}^v \beta_e\mathbf{U}^e\mathbf{Q}^e}{\sum_{e=1}^v \beta_e}, \quad \beta_e = -\log\left(\frac{\delta_e}{\sum_{i=1}^v \delta_i}\right) \tag{15}$$

$$Q_{k,k}^e = \sum_{i=1}^{n_e}\frac{V_{i,k}^e}{\|\mathbf{F}^e\|_F}$$

Algorithm 2 ClusType

Вход: $\boldsymbol{\mathcal{G}}_Q, \boldsymbol{\mathcal{G}}_L, \boldsymbol{\mathcal{G}}_R$,

$\boldsymbol{\mathcal{W}}_{\mathcal{M}}, \boldsymbol{\mathcal{W}}_L, \boldsymbol{\mathcal{W}}_R$,$\mathbf{F}_{context}, \mathbf{F}_{character}$,

$\boldsymbol{\mathcal{Y}}_0$, γ, μ, α, число кластеров k

Выход: $\boldsymbol{\mathcal{Y}}$

$\{\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{P}}_L, \boldsymbol{\mathcal{P}}_R\} \leftarrow \{\boldsymbol{\mathcal{Y}}_0, \boldsymbol{\mathcal{G}}_Q^T\boldsymbol{\mathcal{Y}}_0, \boldsymbol{\mathcal{G}}_L^T\boldsymbol{\mathcal{Y}}_0, \boldsymbol{\mathcal{G}}_R^T\boldsymbol{\mathcal{Y}}_0\}$

$\{\mathbf{U}^e, \mathbf{V}^e, \beta^e\}, \mathbf{V}^* \leftarrow$ положительные числа

повторять

  обновляем $\{\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{P}}_L, \boldsymbol{\mathcal{P}}_R\}$ с помощтю (9) - (12)

  для $e = 1, \ldots, v$

    повторять

      обновляем $\mathbf{V}$ с помощью (13)

      $\mathbf{U} \leftarrow \mathbf{UQ}$, $\mathbf{V} \leftarrow \mathbf{VQ}^{-1}$

      обновляем $\mathbf{U}$ с помощью (14)

    пока δ_e не сойдется

  обновляем $\mathbf{U}^*$, $\{\beta_e\}$ с помощью (15)

пока $\mathcal{O}_{\gamma,\mu,\alpha}$ не сойдется

Выделение именованных сущностей

Опишем в главе способ выделения именованных сущностей.

На первом этапе необходимо построить словарь именованных сущностей. Для этого будем использовать статистический подход к отбору n-грамм, которые могут являться именованными сущностями. Воспользуемся алгоритмом **TopMine** [22]. Алгоритм итеративно конструирует n-граммы слов в предложении основываясь на их частотных характеристиках, процесс останавливается, когда величина значимости для n-граммы становится меньше некоторого заданного порога. Пусть n-грамма ϕ состоит из последовательности слов $w_1, \ldots, w_k$. Через $\nu(\phi)$ – обозначим частоту вхождения n-граммы ϕ в корпус документов $\mathcal{D}$.

Функция значимости алгоритма **TopMine**:

$$\rho\left(\phi_1, \phi_2\right) = \frac{\nu\left(\phi_1 \oplus \phi_2\right) - \nu(\phi_1)\nu(\phi_2)/D}{\sqrt{\nu\left(\phi_1 \oplus \phi_2\right)}} \tag{16}$$

$\oplus$ – конкатенация n-грамм, D – размер коллекции слов. Функция значимости $\rho(\cdot)$ – обеспечивает неслучайность образования n-граммы, а итеративность процесса конструирования n-грамм и фильтрация по порогу – полноту по включению. Алгоритм **TopMine** позволяет находить в корпусе только частотные n-граммы, но именованная сущность, не обязательно обладает высокой частотностью. В корпусах FactRuEval и LABINFORM больше половины размеченных именованных сущностей встречаются в всего один раз. Требуется ввод дополнительных эвристик для выделения низкочастотных n-грамм. Для классов Персона, Организация и Геолокация, n-граммы отличаются еще и написанием с заглавной буквы. Дополнительно будем выделять n-граммы с помощью регулярных выражений, фильтрующих написание слов.

Результаты экспериментов

Для проверки работоспособности предложенного алгоритма был проведен эксперимент на объединении корпусов с конференции FactRuEval и LABINFORM. В документах выделены Персоны, Организации и Геолокации.

- размер объединенного корпуса $\sim$ 300000 слов,
- словарь именованных сущностей:
  - **Персоны**: $\sim$ 6000 n-грамм,
  - **Организации**: $\sim$ 4000 n-грамм,
  - **Геолокации**: $\sim$ 2000 n-грамм,
- именованных сущностей в корпусе:
  - **Персоны**: $\sim$ 12630 n-грамм,
  - **Организации**: $\sim$ 10514 n-грамм,
  - **Геолокации**: $\sim$ 8078 n-грамм,

Качетсво работы предложенных алгоритмов будем измерять с помощью метрик **Precicsion**, **Recall**, **F1 score**.

Таблица 1: Результаты применения алгоритма выделения именованных сущностей.

Precicion	**Recall**	**F1**
0.87	0.92	0.88

Больший вклад, в качество выделения именованных сущностей вносят регулярные выржения с фильтрацией по стилистике написания, так как в корпусе достаточно низкая частотность именованных сущностей и большая часть n-грамм именованных сущностей начинается с заглавных букв.

Цели экспериментов

- Изучение зависимости качества распознавания от размера начальной разметки.

- Изучение зависимости качества распознавания от числа кластеров на множестве связей $\mathcal{R}p$.

Были проведены серии испытаний, в которых на кажом испытании случайным образом выбиралось подмножество $\mathcal{M}_0$ разных размеров, и разное число кластеров. При проведении экспериментов для связей $r \in \mathcal{R}p$ выделялись следующие признаки. Выбирались все уникальные слова из предложений s, в которых встречалась каждая из связей r, а также из предыдущего s_{-1} и следующего s_{+1} предложений. В результате был сформирован словарь контекста $\{w_1^{context}, \ldots, w_{n_s}^{context}\}$. Каждое слово взвешивалось по tf-idf. В итоге была сформирована матрица объект-признак $\mathbf{F}_{context} \in \mathbb{R}^{|\mathcal{R}p| \times n_s}$. Матрица $\mathbf{F}_{character}$ в эксперименте не использовалась. Аналогично связям строились признаки для $m \in \mathcal{M}$ и в качестве $\boldsymbol{\mathcal{W}}_{\mathcal{M}}$ был выбран KNN-граф с heat-kernel функцией весов на ребрах [23]. Двудольные графы $\boldsymbol{\mathcal{G}}_L, \boldsymbol{\mathcal{G}}_R$ строились исходя из взаимного расположения в предложениях m и r. Все столбцы матриц двудольных графов $\boldsymbol{\mathcal{G}}_{\mathcal{Q}}, \boldsymbol{\mathcal{G}}_L, \boldsymbol{\mathcal{G}}_R$ были нормализованы по ℓ_2 норме, для того, чтобы уменьшить влияние связей и именованных сущностей с высокой частотностью в корпусе на результаты работы алгоритма.

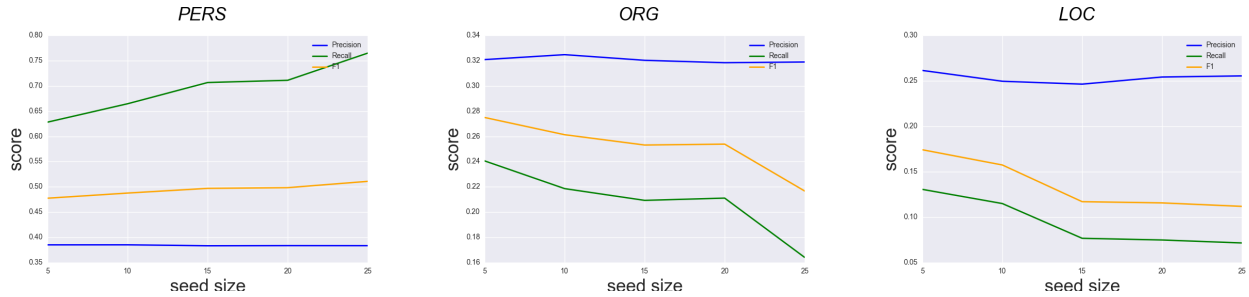

Рисунок 1: 10 кластеров

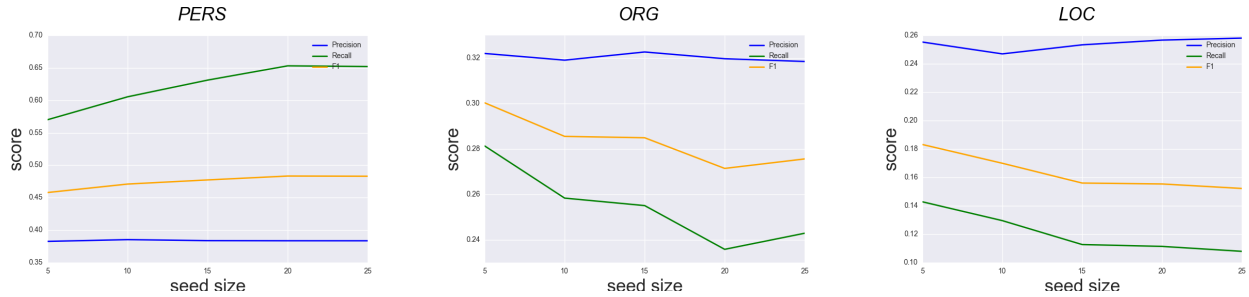

Рисунок 2: 100 кластеров

При увеличении размера начальной выборки ожидается увеличение качества распознавания для всех классов, однако, графики на рисунках 1,2 показывают, что линия Precision для всех классов отстается неизменной и увеличивается только Recall для класса Персона, при этом Recall для всех остальных классов уменьшается, причем эта динамика не зависит от числа кластеров. Это может означать, что алгоритм при отдает предпочтение объекту

класса Персоны. Связано это может быть с тем что, двудольные графы имеют локальную область сгущения и ребра в этой области связаны с вершинами графов относящимся к Персонам. Об этом также говорят результаты в таблице 2, так как кластеризация увеличивает Recall для класса Персона, при этом уменьшая Recall для всех остальных классов.

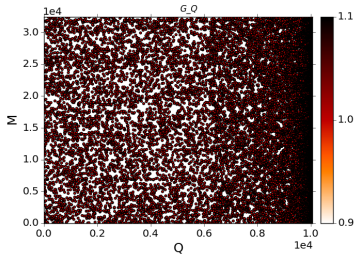

Рисунок 3: $\mathcal{G}_{\mathcal{Q}}$

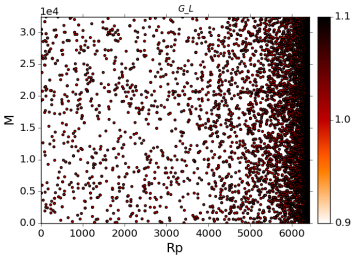

Рисунок 4: $\mathcal{G}_L$

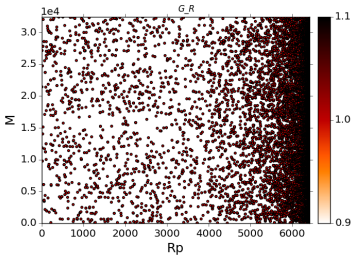

Рисунок 5: $\mathcal{G}_R$

Таблица 2: Результаты на корпусе FactRuEval и LABINFORM

Алгоритм	**Person**			**Location**			**Organization**		
	Precision	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
CLUS	0.38	**0.65**	**0.48**	0.25	0.10	0.15	0.32	0.24	0.27
NOCLUS	0.39	0.54	0.45	0.25	**0.16**	**0.2**	**0.32**	**0.32**	**0.32**

Заключение

В работе предложена модель для автоматического выделения именованных сущностей в коллекциях текстовых документов. Проведен эксперимент на размеченном корпусе на русском языке. Алгоритм показывает низкое качество распознавания на корпусе текстов малого объема.

План дальнейших работ

- Провести эксперименты на расширенном корпусе текстов.
- Исследовать динамику изменения качества при увеличении выборки.
- Сравнить работу алгоритма с существующими решениями для русского языка.
- Попытаться обобщить алгоритм для использования на корпусах текстов других тематик.

Литература

1. Bikel Daniel M., Schwartz Richard, Weischedel Ralph M. An Algorithm that Learns What's in a Name // Machine Learning. 1999. Т. 34, № 1. С. 211–231.

2. Sutton Charles, McCallum Andrew. An Introduction to Conditional Random Fields // Found. Trends Mach. Learn. 2012. Т. 4, № 4. С. 267–373.

3. Bender Oliver, Och Franz Josef, Ney Hermann. Maximum Entropy Models for Named Entity Recognition // Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003. Т. 4 из *CONLL '03*. Association for Computational Linguistics, 2003. С. 148–151.

4. Lafferty John D., McCallum Andrew, Pereira Fernando C. N. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data // Proceedings of the Eighteenth International Conference on Machine Learning. ICML '01. Morgan Kaufmann Publishers Inc., 2001. С. 282–289.

5. Efficient Estimation of Word Representations in Vector Space / Tomas Mikolov, Kai Chen, Gregory S. Corrado [и др.] // CoRR. 2013. Т. abs/1301.3781.

6. Pennington Jeffrey, Socher Richard, Manning Christopher D. Glove: Global vectors for word representation // In EMNLP. 2014.

7. Santos Cicero Dos, Zadrozny Bianca. Learning Character-level Representations for Part-of-Speech Tagging // Proceedings of the 31st International Conference on Machine Learning. Т. 32. PMLR, 2014. С. 1818–1826.

8. Joint Learning of Character and Word Embeddings / Xinxiong Chen, Lei Xu, Zhiyuan Liu [и др.] // Proceedings of the 24th International Conference on Artificial Intelligence. AAAI Press, 2015. 7 с.

9. Anh L. T., Arkhipov M. Y., Burtsev M. S. Application of a Hybrid Bi-LSTM-CRF model to the task of Russian Named Entity Recognition // CoRR. 2017. Т. abs/1709.09686.

10. Liao Wenhui, Veeramachaneni Sriharsha. A Simple Semi-supervised Algorithm for Named Entity Recognition // Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing. Association for Computational Linguistics, 2009. C. 58–65.

11. Kuksa Pavel P., Qi Yanjun. Semi-supervised Bio-named Entity Recognition with Word-Codebook Learning // SDM. 2010.

12. Li Lishuang, Zhou Rongpeng, Huang Degen. Two-phase biomedical named entity recognition using CRFs // Computational biology and chemistry. 2009. T. 33 4. C. 334–8.

13. Clustype: Effective entity recognition and typing by relation phrase-based clustering / Xiang Ren, Ahmed El-Kishky, Chi Wang [и др.] // Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining / ACM. 2015. C. 995–1004.

14. Chapelle Olivier, Schlkopf Bernhard, Zien Alexander. Semi-Supervised Learning. 1st изд. The MIT Press, 2010.

15. Lee Daniel D., Seung H. Sebastian. Algorithms for Non-negative Matrix Factorization // Proceedings of the 13th International Conference on Neural Information Processing Systems. NIPS'00. MIT Press, 2000. C. 535–541.

16. Gonzalez Edward F., Zhang Yin. Accelerating the Lee-Seung Algorithm for Nonnegative Matrix Factorization. 2005.

17. Riabenko Evgeniy. Multiplicative method for non-negative matrix factorization with AB-divergence and its convergence. 2014. 01. T. 1. C. 800–816.

18. Ding Chris, Li Tao, Peng Wei. On the Equivalence Between Non-negative Matrix Factorization and Probabilistic Latent Semantic Indexing // Comput. Stat. Data Anal. 2008. T. 52, № 8. C. 3913–3927.

19. Gaussier Eric, Goutte Cyril. Relation Between PLSA and NMF and Implications // Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '05. ACM, 2005. C. 601–602.

20. Multi-View Clustering via Joint Nonnegative Matrix Factorization / Jing Gao, Jiawei Han, Jialu Liu [и др.] // SDM. 2013. C. 252–260.

21. Wu Siyuan, Wang Jim. Nonnegative Matrix Factorization: When Data is not Nonnegative // 7th International Conference on BioMedical Engineering and Informatics. 2014.

22. Scalable Topical Phrase Mining from Text Corpora / Ahmed El-Kishky, Yanglei Song, Chi Wang [и др.] // Proc. VLDB Endow. 2014. Т. 8, № 3. С. 305–316.

23. He Xiaofei. Locality Preserving Projections. Ph.D. thesis. University of Chicago, 2005.