УДК 004.021

И. А. Литвинович, А. С. Наркевич
Белорусский государственный технологический университет

РАЗРАБОТКА И ОПТИМИЗАЦИЯ АЛГОРИТМОВ ПОИСКА ПРОФИЛЕЙ ПОЛЬЗОВАТЕЛЕЙ СОЦИАЛЬНОЙ СЕТИ ПО ФОТОГРАФИИ

В статье рассмотрены алгоритмы определения принадлежности профиля пользователя социальной сети к одному из лиц, найденному на фотографиях данного профиля. Этот процесс называется построением пользовательского профиля и направлен на нахождение однозначного соответствия лица профилю. Также в статье приведен поэтапный обзор разработанного алгоритма построения пользовательского профиля. Данный алгоритм основывается на использовании сверточной нейронной сети FaceNet для обработки фотографий и нахождения лиц, а также алгоритма кластеризации найденных лиц. Узким местом разработанного алгоритма построения профилей является кластеризация. Для получения максимальной производительности и точности были исследованы несколько популярных алгоритмов кластеризации. Сделан обзор самых популярных алгоритмов, произведены замеры производительности и надежности каждого из них. В качестве наиболее оптимального выбран алгоритм DBSCAN. Область применения алгоритма построения пользовательского профиля, описанного в статье, достаточно обширна, однако основной целью является подготовка набора пользовательских данных для последующего поиска пользовательских профилей по фотографии в определенной социальной сети. Разработанный алгоритм был успешно применен и хорошо себя зарекомендовал.

Ключевые слова: распознавание лиц, кластеризация, алгоритм, социальная сеть, характеристический вектор.

I. A. Litvinovich, A. S. Narkevich
Belarusian State Technological University

DEVELOPMENT AND OPTIMIZATION OF ALGORITHMS FOR SEARCHING PROFILES OF SOCIAL NETWORK USERS BY PHOTO

This article discusses algorithms for determining whether a profile of a social network user belongs to one of the individuals found in the photos of this profile. This process is called building a user profile and is aimed at finding a one-to-one correspondence of a person to a profile. This article also provides a step-by-step overview of the developed algorithm for constructing a user profile. This algorithm is based on the use of the FaceNet convolutional neural network for processing photos and finding faces, as well as the algorithm for clustering found faces. The bottleneck of the developed algorithm for constructing profiles is clustering. To obtain maximum performance and accuracy, several popular clustering algorithms were investigated. A review of the most popular algorithms is made, the performance and reliability of each of them are measured. The DBSCAN algorithm was chosen as the most optimal. The scope of the algorithm for constructing a user profile described in this article is quite extensive, however, the main goal is to prepare a set of user data for the subsequent search for user profiles from a photograph in a certain social network. The developed algorithm was successfully applied and proved to be good in the next stage of my work.

Key words: face recognition, clustering, algorithm, social network, characteristic vector.

Введение. Основная цель работы – это точное определение принадлежности профиля одному из лиц, найденных на фотографиях. Профиль в социальной сети – это личная страница зарегистрированного пользователя с указанием личной информации о нем, включая фото профиля (или аватар − пользовательское фото), сведения о друзьях, статусах, группах, сообществах, записи на стене, фотографии и пр. Основная сложность определения однозначного соответствия лица профилю заключается в том, что пользователи социальных сетей часто выкладывают групповые фотографии, а также фотографии знаменитостей, друзей, кумиров, а также фотографии, на которых людей нет. Для достижения цели важно решить две задачи:

1) выбор критериев, по которым необходимо отбирать фотографии для обработки;

2) выбор максимально эффективного алгоритма кластеризации лиц, найденных на фотографиях.

Кластеризация – это процесс объединения объектов с одинаковыми характеристиками в одну группу, а с различными характеристиками в другие группы. После отбора необходимых фотографий, проведения обработки данных фотографий и кластеризации результатов следует вычислить центроид лидирующего кластера,

который определяется на основании векторов, входящих в него. Центроид – точка, которая является центром кластера. В итоге объекты в одном кластере имеют схожие характеристики, что означает, что отдельный человек попадает в свой кластер.

Основная часть. В общем случае алгоритм построения пользовательского профиля содержит три этапа, каждый из которых имеет свои входные и выходные данные.

На первом этапе выполняется скачивание фотографий из пользовательского аккаунта. Входом для данного этапа является список всех фотографий пользователя, включая всю метаинформацию, такую как разрешение фотографий, количество лайков, количество комментариев, является ли фотография аватаркой.

На втором этапе происходит нахождение всех лиц на уже выбранных на предыдущем этапе фотографиях. Для распознавания лиц используется нейронная сеть FaceNet. FaceNet – это технология Google, опубликованная в 2015 г., разработанная Ф. Шрофом, Д. Калиниченко и Д. Филибином и описанная в [1]. Для распознавания предназначена обученная глубокая сверточная нейронная сеть, которая возвращает 128-размерный вектор признаков, отлично классифицирующийся. Евклидово расстояние в 128-мерном пространстве используется в качестве критерия для измерения похожести лиц.

На третьем этапе производится кластеризация всех найденных на предыдущем этапе векторов для получения некоторого количества кластеров, из которых по определенным признакам выбирается лидирующий кластер и вычисляется его центроид, представляющий собой вектор, который характеризует все лица из заданного кластера. Лидирующий кластер – это тот кластер, лица на котором принадлежат владельцу профиля. Лидирующий кластер определяется на основании размера кластера, т. е. лидирующим выбирается кластер с наибольшим количеством входящих в него векторов. После получения единого вектора (центроида) он сохраняется в базе данных с привязкой к текущему пользовательскому аккаунту. Этот вектор может быть использован как векторное представление лица владельца аккаунта.

Кластеризация применяется к математическим (или в данном случае векторным) представлениям всех лиц. Различия в этих векторах определяются на основании расстояния между точками, которым соответствуют векторы. Расстояние можно рассчитать, взяв евклидово расстояние между двумя векторами. Евклидово расстояние между двумя точками, которым соответствуют радиус-векторы $p = \{p_1, p_2, ..., p_{128}\}$ и $q = \{q_1, q_2, ..., q_{128}\}$, имеющие 128 компонент, вычисляется следующим способом:

$$d(p,q) = \sqrt{\sum_{i=1}^{128} (q_i - p_i)^2}.$$

Для кластеризации существует несколько подходов с различной производительностью. Цель этого исследования – оценка различных алгоритмов кластеризации, описанных ниже.

Пороговая кластеризация – это подход к кластеризации, предложенный в [1]. Добавление нового вектора в кластер оценивается на основании уже кластеризованных векторов с учетом расстояния между двумя точками, которые ему соответствуют. Если расстояние между новым лицом и его ближайшим соседом в наборе, лицо которого уже кластеризовано, меньше порогового значения, заданного пользователем, то лицо добавляется в существующий кластер. Если для данного лица все расстояния ниже порога, т. е. совпадений нет, то необходимо создать новый кластер. Пороговое значение играет важную роль в этом подходе. Выбор низкого порогового значения приводит к множеству ложных негативных срабатываний: пара лиц, которые имеют расстояние выше порога, но получены от разных людей. Поэтому во время эксперимента крайне важно осторожно указывать этот параметр, чтобы получить желаемый результат.

В методе кластеризации Mean Shift, описанном Команичиу и Меером [2], каждый характеристический вектор представлен в евклидовом пространстве. Основное распределение оценивается с помощью подхода, называемого оценкой плотности ядра. Это работает путем размещения ядра в каждой точке набора данных и перемещения каждой точки в направлении ее изменения. На примере кандидата x_i правило обновления для итерации t выглядит следующим образом:

$$x_i^{t+1} = x_i^t + m(x_i^t),$$

где $m(x_i)$ – средний вектор сдвига, который вычисляется для каждого лица и указывает на область максимального увеличения плотности точек:

$$m(x_i) = \frac{\sum_{x_j \in N(x_i)} K(x_j - x_i) x_i}{\sum_{x_j \in N(x_i)} K(x_j - x_i)},$$

здесь $N(x_i)$ – окрестность выборок на заданном расстоянии вокруг x_i; функция $K(x_j - x_i)$ определяет вес каждого элемента.

Преимущество этого подхода состоит в том, что он является непараметрическим алгоритмом, так как он не делает предположений о данных.

Например, (в отличие от *k*-средних) количество кластеров (или прототипов) не указывается.

Пространственная кластеризация данных с шумом на основе плотности Density-Based Spatial Clustering of Application with Noise (DBSCAN) – это алгоритм кластеризации данных, предложенный М. Эстером, Х.-П. Кригелем, Ю. Сандером и С. Сюй в 1996 г. Это также непараметрический подход. Не нужно заранее указывать точное количество кластеров. Вместо этого, учитывая набор точек данных (или вложений), DBSCAN группирует точки, которые лежат близко друг к другу на основе евклидова расстояния. Алгоритм DBSCAN требует двух параметров: минимального расстояния между двумя точками, которые могут быть сгруппированы вместе, и минимального расстояния для формирования плотной области. В случае кластеризации лиц минимальное количество точек для формирования плотной области должно быть равно 1: если это число равно 1, то лицо без близких соседей может образовывать кластер с одним элементом. Расстояние между двумя точками, которые можно сгруппировать, может варьировать, и наилучшее значение должно быть получено во время экспериментов.

Для сравнения подходов кластеризации был выбран набор данных под названием Labeled Faces in the Wild (LFW), собранный исследователями из Университета Массачусетса. Этот набор данных позволяет тестировать методы в маркированной базе данных. Маркировка означает, что личность человека на изображении известна. LFW содержит 13 233 изображения 5749 человек, где 1680 человек имеют два или более изображений, остальные имеют не более одного.

Для сравнения производительности различных алгоритмов кластеризации могут быть использованы различные оценки. Одной из таких мер качества кластеров является попарная *F*-мера, которая использовалась в работе.

Введем следующие определения. Рассмотрим два набора меток: L и C. Набор $L = \{l_1, l_2, ..., l_n\}$ содержит фактические метки для каждого лица, используемого в кластеризации. Набор $C = \{c_1, c_2, ..., c_n\}$ является выходом алгоритма кластеризации для каждого лица.

Количество истинных позитивных срабатываний (*TP*) или чувствительности состоит из пар лиц (i, j), которые правильно сгруппированы в один кластер. *TP* определяется как:

$$TP = |(i, j)|, \text{ где } c_i = c_j \text{ и } l_i = l_j.$$

Количество ложных срабатываний (*FP*) состоит из пар лиц (i, j), которые неправильно сгруппированы в одном кластере. Количество ложных срабатываний вычисляется как:

$$FP = |(i, j)|, \text{ где } c_i = c_j \text{ и } l_i \neq l_j.$$

Количество позитивных срабатываний при ожидаемом негативном (*TN*) состоит из пар, которые сгруппированы в кластер, хотя должны находиться в разных кластерах. Количество истинных негативных срабатываний рассчитывается как:

$$TN = |(i, j)|, \text{ где } c_i \neq c_j \text{ и } l_i \neq l_j.$$

Количество ложных негативных срабатываний (*FN*) состоит из пар граней (i, j), которые неправильно сгруппированы в разные кластеры. Количество ложных негативных срабатываний находится как:

$$FN = |(i, j)|, \text{ где } c_i \neq c_j \text{ и } l_i = l_j.$$

Попарная точность (*P*) определяется как отношение пар, которые правильно сгруппированы в один и тот же кластер (*TP*), ко всем парам, которые фактически были сгруппированы в один и тот же кластер с помощью алгоритма кластеризации (*TP* + *FP*). Поэтому попарная точность вычисляется как:

$$P = \frac{TP}{TP + FP}.$$

Полнота (*R*) – это доля пар, которые правильно сгруппированы в один кластер (*TP*) по всем парам одного кластера (*TP* + *FN*). Полнота рассчитывается как:

$$R = \frac{TP}{TP + FN}.$$

F-мера (*F*-measure) – характеристика, которая позволяет дать оценку одновременно по точности и полноте:

$$F = 2\frac{P \cdot R}{P + R}.$$

F-мера эффективно определяет окончательную кластеризацию, выполняемую алгоритмом кластеризации. Если алгоритм кластеризации создает единый кластер для каждого отдельного лица, точность высокая, но полнота крайне низкая. В этом случае *F*-мера задает низкую оценку производительности. Если алгоритм кластеризации создает один кластер, содержащий все лица, полнота высокая, но точность низкая. *F*-мера также указывает на плохую производительность в этом случае.

Одна из целей кластеризации – кластеризация с осторожностью, т. е. желательно кластеризовать лица только тогда, когда есть уверенность, что изображения содержат одного и того же человека. Нежелательный эффект *F*-меры заключается в том, что он может увеличиваться при росте количества ложных срабатываний.

Так как целесообразно уменьшить количество ложных срабатываний, поэтому нужно установить лимит ложных срабатываний; он не может быть больше чем 1% от количества изображений в наборе данных.

F-мера получается путем оценки истинных или ложных положительных результатов и истинных или ложных отрицательных значений.

Каждый алгоритм кластеризации имеет определенный параметр, который может варьировать, что приводит к различным результатам кластеризации. Для получения значений в таблице использовались следующие параметры:

– пороговая кластеризация: порог – 0,49 ед.;

– среднее смещение: пропускная способность – 0,38 ед.;

– DBSCAN: расстояние – 0,40 ед.

Результаты работы алгоритмов

Метод	*F*, ед.	Количество, шт.	*FP*, ед.	*P*, ед.	*R*, ед.
Разбиение вручную	1,0	4935	0	1,0	1,0
Mean Shift	0,12	4701	10	0,95	0,06
DBSCAN	0,14	4850	7	0,99	0,08
Пороговая кластеризация	0,13	4840	11	0,98	0,08

Заключение. В наборе данных Labeled Faces in the Wild алгоритм кластеризации DBSCAN показал сопоставимую производительность с алгоритмом Mean Shift. Для большого набора данных, использованного в эксперименте, производительность аналогична Mean Shift, но кластеризация DBSCAN, исходя из результатов, приведенных в таблице, лучше с точки зрения точности, полноты и небольшого количества ложных срабатываний. Также DBSCAN продемонстрировал лучшую *F*-меру (0,14 ед.), чем пороговая кластеризация (0,13 ед.). Учитывая это, делается вывод, что DBSCAN является алгоритмом кластеризации, который хорошо работает при кластеризации лиц и, следовательно, является приоритетным при выборе алгоритма кластеризации для построения пользовательского профиля.

Литература

1. Schroff F., Kalenichenko D., Philbin J. FaceNet: A Unified Embedding for Face Recognition and Clustering // CoRR abs/1503.03832 (2015). arXiv: 1503.03832. URL: http://arxiv.org/abs/1503.03832 (date of access: 02.10.2019).

2. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments / Gary B. Huang [et al.]. Amherst: University of Massachusetts, 2007. P. 1–10.

References

1. Schroff F., Kalenichenko D., Philbin J. FaceNet: A Unified Embedding for Face Recognition and Clustering. *CoRR abs/1503.03832 (2015). arXiv: 1503.03832*. Available at: http://arxiv.org/abs/1503.03832 (accessed 02.10.2019).

2. Gary B. Huang, Ramesh M., Berg T., Learned-Miller E. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Amherst, University of Massachusetts, 2007. P. 1–10.

Информация об авторах

Литвинович Игорь Алексеевич – магистрант. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: ihar.litvinovich@ gmail.com

Наркевич Аделина Сергеевна − старший преподаватель кафедры программной инженерии. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: nas@belstu.by

Information about the authors

Litvinovich Ihar Alekseevich – Master's degree student. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: ihar.litvinovich@gmail.com

Narkevich Adelina Sergeevna − Senior Lecturer, the Department of Software Engineering. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: nas@belstu.by

Поступила после доработки 13.11.2019