

Многоклассовая классификация в задаче семантической сегментации

Sobolev Alexander, Varinova Olga

Факультет Вычислительной Математики и Кибернетики

Московский Государственный Университет им. Ломоносова, Москва, Россия

alexandersobolev@yahoo.com

Аннотация

Данная работа посвящена рассмотрению существующих методов многоклассовой классификации на основе бинарного классификатора AdaBoost и усовершенствованию алгоритмически простого подхода «один против всех» с целью улучшения качества его работы в терминах значений ошибки на тестовых наборах данных в задаче семантической сегментации.

Ключевые слова: многоклассовая классификация, семантическая сегментация, «один против всех», AdaBoost, Platt Scaling.

1. ВВЕДЕНИЕ

Пусть I – произвольное цифровое изображение, состоящее из пикселей $x \in I$; $\{1, 2, \dots, K\}$ – множество из K классов визуальных объектов (например, {Шум, Земля, Здание, Человек, Небо}). Тогда задача семантической сегментации сводится к построению алгоритма, реализующего отображение $f^*: \{x|x \in I\} \rightarrow \{1, 2, \dots, K\}$, согласующееся с реальным визуальным расположением объектов на изображении. Другими словами, необходимо построить алгоритм, приписывающий каждому пикселю входного изображения одну из предопределенных меток класса согласно реальному нахождению объекта на изображении.

Таким образом, при решении задачи семантической сегментации, самой сегментации (то есть разбиения изображения на односвязные в дискретном смысле непересекающиеся множества пикселей, в объединении составляющие все изображение) может и не происходить. Тогда итоговыми сегментами можно считать наибольшие односвязные в дискретном смысле множества одинаково размеченных пикселей изображения.

Решения задачи семантической сегментации часто используют методы многоклассовой классификации.

Пусть задано K классов с соответствующими метками $\{1, 2, \dots, K\}$, пространство \mathbb{R}^m векторов признаков – числовых описаний объектов и априорно размеченная тренировочная выборка $S \subset \mathbb{R}^m \times \{1, 2, \dots, K\}$ – конечное множество известных пар вида {Вектор признаков, Метка класса}. А также, пусть $x \in \mathbb{R}^m$ – некоторый вектор признаков, обязательно принадлежащий тренировочной выборке. Предполагается, что существует некоторое неизвестное отображение $f: \mathbb{R}^m \rightarrow \{1, 2, \dots, K\}$, согласно которому размечена тренировочная выборка S . Тогда задача многоклассовой классификации состоит в построении по тренировочной выборке S алгоритма, реализующего функцию $f^*: \mathbb{R}^m \rightarrow \{1, 2, \dots, K\}$ такую, чтобы $E(\|f^*(x) - f(x)\|)$ было минимальным. Стоит отметить, что x в данном случае не обязательно должен быть из тренировочной выборки S . Другими словами по

тренировочной выборке необходимо построить функцию f^* , обобщающую неизвестную зависимость векторов признаков и меток классов из S .

В данной статье в разделе 2 приводится обзор методов многоклассовой классификации, основанных на идее усиления бинарных гипотез AdaBoost [1]. В разделе 3 рассмотрен предложенный метод многоклассовой классификации и проведена оценка сложности классификации рассмотренными алгоритмами. В разделе 4 приведено экспериментальное сравнение рассмотренных методов на примере работы в реальной системе семантической сегментации.

2. ОБЗОР МЕТОДОВ МНОГОКЛАССОВОЙ КЛАССИФИКАЦИИ

Данный обзор фокусируется на методах разрешения проблем многоклассового машинного обучения, основанных на идее усиления слабых гипотез AdaBoost [1].

Большинство существующих методов многоклассовой классификации либо базируются на бинарных классификаторах, либо к ним сводятся. Общая идея такого сведения заключается в использовании набора бинарных классификаторов (не обязательно AdaBoost), обученных разделять различные группы объектов друг от друга. При классификации используются различные схемы голосования.

В общем, все современные многоклассовые методы машинного обучения можно подразбить на 3 группы:

1. Элементарные – простые подходы, использующие набор бинарных классификаторов «один против всех» и «каждый против каждого» [2];
2. Классические – методы, сводимые к группе элементарных подходов (AdaBoost.MH, AdaBoost.M2, AdaBoost.MR [1]);
3. Методы, основанные на кодах с коррекцией ошибки (ECC – error correcting code). К ним относятся: AdaBoost.ECC [3], AdaBoost.ERP [4], AdaBoost.ERC [5].

Метод «один против всех» состоит в построении k бинарных классификаторов h_i , каждый из которых обучается на тренировочной базе вида $\{(x, 1)|y = i, (x, y) \in S\} \cup \{(x, -1)|y \neq i, (x, y) \in S\}$, $i = 1, k$, где i – порядковый номер бинарного классификатора. Классификация произвольного вектора признаков $x \in \mathbb{R}^m$ определяется голосованием: $F(x) = \operatorname{argmax}_i (h_i(x))$.

В статье «Statistical view of boosting» [1] показано, что группа «классических» методов сводится к методу «один против всех».

Основной идеей методов, основанных на самокорректирующихся кодах, состоит в кодировке меток классов уникальными бинарными словами одинаковой

длины, которые образуют кодирующую матрицу M (ее строки совпадают с кодируемыми словами). Далее для каждого столбца строится бинарный классификатор, умеющий отделять классы с соответствующим значением в столбце, равным единице, от классов с нулем.

В сравнении классификаторов AdaBoost.ECC, ERP и ERC [5] было показано, что AdaBoost.ERP и ERC – справляются с задачей классификации значительно лучше, чем классический метод «один против всех».

3. ОПИСАНИЕ ПРЕДЛОЖЕННОГО МЕТОДА

Для решения поставленной задачи проанализируем простейший метод «один против всех» и усовершенствуем его, не сильно усложнив процесс классификации.

3.1 Улучшение метода «один против всех»

Классификация в методе «один против всех» происходит следующим образом: $\mathcal{F}(x) = \operatorname{argmax}_i (h_i(x))$. Но из схемы настройки классификатора ясно, что все бинарные классификаторы $h_i(x)$ настраиваются каждый независимо друг от друга.

Бинарные классификаторы на выход выдают некоторый аналог расстояния до построенной ими разделяющей поверхности (так, в случае слабых учеников – порогов по одной из компонент векторов признаков – выход бинарного Gentle AdaBoost соответствует L1-расстоянию до разделяющей классы поверхности), а не точную оценку вероятности принадлежности вектора x 1-ому классу.

Таким образом, в методе «один против всех» бинарные классификаторы получаются несогласованными, и встает вопрос о «согласовании» их выходов.

Более того, в подходе «один против всех» встает проблема несбалансированности классов. Так, даже если изначально задача многоклассовой классификации сбалансирована (то есть имеет одинаковое число тренировочных прецедентов каждого класса), то при обучении бинарного классификатора прецедентов с меткой 1 будет N/K , а прецедентов с меткой -1 будет $N(N-1)/K$. Таким образом, отношение количества прецедентов в каждой бинарной задаче будет равно $(N-1)$, а оно увеличивается с ростом числа классов, что будет особенно сильно сказываться на задаче семантической сегментации с большим числом классов.

Существует метод масштабирования или шкалирования выходов $\mathcal{F}(x)$ уже настроенных бинарных классификаторов – Platt Scaling [6]. Он преобразовывает выходы бинарных классификаторов в оценку вероятности принадлежности прецедента x к 1-ому классу, т.е. в оценку $P(1|x)$. Для работы самого метода необходима так называемая валидационная выборка (для нее априорно известны метки классов и при этом классификатор не настраивался на ней). Ее часто создают из обучающей выборки, немного сокращая последнюю.

Улучшенный метод «один против всех»

Задача с K классами, обучающей выборкой D .

Для каждого класса $k = \overline{1, K}$ выполнить:

1. Случайным образом проредить или продублировать данные D_k вида $D_k = \{(x, 1) | y = k, (x, y) \in D\} \cup \{(x, -1) | y \neq k, (x, y) \in D\}$
2. Выделить валидационную выборку V_k , как часть D_k
3. Настроить бинарный классификатор h_k на данных $D_k \setminus V_k$

4. Получить с помощью Platt Scaling сигмоиду $S_k = \{A_k, B_k\}$ по выходам классификатора h_k на данных V_k

Выход классификатора:

$$\operatorname{argmax}_k S_k(h_k(x)) = \operatorname{argmax}_k \frac{1}{1 + e^{A_k h_k(x) + B_k}}$$

Алгоритм 1: *улучшенный метод «один против всех».*

Для борьбы с несбалансированностью классов можно использовать простейшие методы случайного прореживания или дублирования данных [7][8].

Улучшим метод «один против всех», определив схему работы согласно Алгоритму 1 (добавив шкалирование по платту к каждому бинарному классификатору и проредим или продублируем данные для решения проблемы несбалансированности классов).

3.2 Оценка сложности классификации различными методами

Пусть все бинарные классификаторы многоклассовых классификаторов имеют тип Gentle AdaBoost и используют фиксированное число T одинаковых слабых учеников. Пусть также число классов равно K . Тогда сложность классификации будем оценивать по числу необходимого количества слабых учеников для работы методов.

Оценим сложность классификации полученным методом в выше определенных терминах. Она составляет $O(K * (2 \text{ операции произведения} + 2 \text{ суммирования} + \text{одно возведение в степень} + T)) = O(K(T + \text{const})) = O(TK)$.

Сравним сложности классификации методом AdaBoost.ECC/ERP и «один против всех».

Утверждение 1 – о сложности классификации методом AdaBoost.ECC/ERP

В общем случае классификация алгоритмом «один против всех» как минимум не сложнее классификации методом AdaBoost.ECC/ERP.

Доказательство.

Сведем классификацию методом «один против всех» к классификации методом AdaBoost.ECC/ERP, чем докажем поставленное утверждение.

Пусть есть конкретный классификатор на основе метода «один против всех». Пусть он состоит из K бинарных классификаторов h_i , $i = \overline{1, K}$ и с T слабыми учениками каждый.

Построим кодирующую матрицу M классификатора AdaBoost.ECC следующим образом: $M = 2 * E^{K \times K} - 1$, где $E^{K \times K}$ – единичная квадратная матрица размерами $K \times K$. Таким образом, кодирующая матрица M представляет собой квадратную матрицу, на главной диагонали которой находятся единицы, а в остальных ячейках – минус единицы.

Определим бинарные классификаторы метода AdaBoost.ECC как $f_t = h_t$, $t = \overline{1, K}$, а веса $\alpha_t = 1$, $t = \overline{1, K}$. Тогда процесс классификации упростится:

$$\begin{aligned} \operatorname{argmin}_k \Delta (M(k), \mathcal{F}(x)) &= \sum_{t=1}^T \alpha_t * \frac{1 - M(k, t) f_t(x)}{2} \\ &= \frac{1}{2} \left\{ K - \sum_{t=1}^K M(k, t) h_t(x) \right\} \leftrightarrow \\ &\leftrightarrow \operatorname{argmax}_k \left\{ \sum_{t=1}^K M(k, t) h_t(x) \right\}. \quad (*) \end{aligned}$$

Не ограничивая общности, пусть $h_1(x) > h_2(x), \dots, h_k(x)$. Тогда в задаче «один против всех» решением является 1 класс. Рассмотрим построенный классификатор AdaBoost.ECC. Сравним значение (*) при $k = 1$ с произвольным $k = i$.

$$\begin{aligned} h_1 - h_2 - \dots - h_i - \dots - h_k &\sqrt{-h_1 - \dots + h_i - \dots - h_k}, \\ 2h_1 &\sqrt{2h_i}, \\ h_1 &> h_i, \forall i \neq 1 \end{aligned}$$

Таким образом, решение конкретной задачи «один против всех» является решением построенной задачи AdaBoost.ECC ($\operatorname{argmax}_k = 1$) Но все указанные рассуждения справедливы и в обратном направлении – решение построенной задачи AdaBoost.ECC является решением задачи «один против всех». Согласно утверждению о сложности алгоритмов из теории математической оптимизации, получаем доказательство представленного утверждения (классификация «один против всех» корректно сводится к классификации AdaBoost.ECC, что говорит о том, что задача классификации AdaBoost.ECC как минимум не проще метода «один против всех»).

Сложность алгоритмов классификации в указанных терминах приведена в Таблице 1.

Таблица 1 – сравнение сложности классификации различными многоклассовыми методами	
Метод	Сложность классификации
«Один против всех» и улучшенный «один против всех»	$O(TK)$
AdaBoost.M1, M2, MN и др.	$\sim O(TK)$
«Каждый против каждого»	$O(TK^2)$
AdaBoost.ECC/ERP	Не проще, чем $O(TK)$

Таким образом, был получен новый метод многоклассовой классификации, незначительно отличающийся по алгоритмической сложности классификации от простого метода «один против всех».

4. ЭКСПЕРИМЕНТЫ

Сравним полученный метод с рассмотренными алгоритмами многоклассовой классификации.

Оценкой качества работы рассмотренных методов будем считать значение ошибки построенных классификаторов на тестовой выборке со скользящим окном в зависимости от числа слабых учеников, входящих в классификатор. Для семантической сегментации под уровнем ошибки подразумевается попиксельная ошибка на тестовой выборке размеченных изображений.

В качестве тестовой системы семантической сегментации был выбран упрощенный вариант системы TextonBoost [9] (попиксельная система семантической сегментации).

В качестве тренировочных и тестовых изображений были взяты базы Sowerby [10] – 5 классов и MSRC21 [11] – 21 класс. Эти базы являются одними из наиболее распространенных для сравнения систем семантической сегментации. В результирующих изображениях все обученные классификаторы состоят из одинакового числа слабых учеников (рис. 1,2).

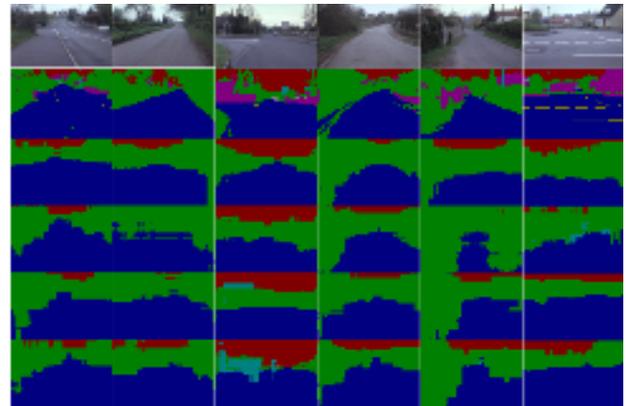


Рис. 1: Результаты работы методов многоклассовой классификации в системе TextonBoost, база Sowerby. Сверху вниз – оригинал изображения, ручная разметка, JointBoost в реализации авторов TextonBoost, AdaBoost.ERP, «один против всех» и собственный улучшенный «один против всех».

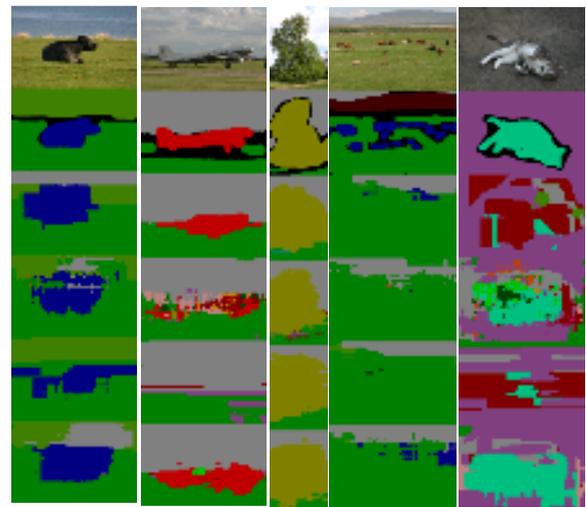
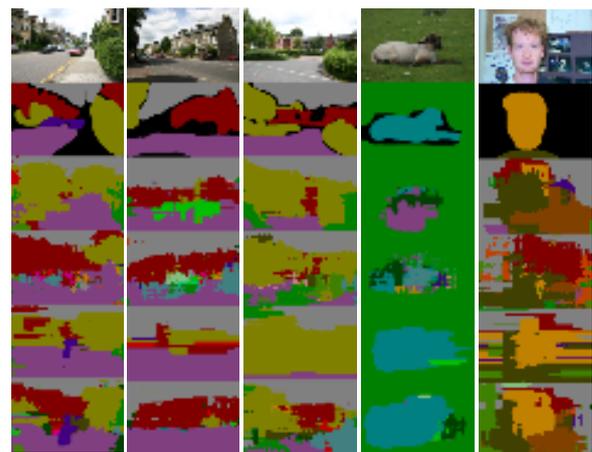


Рисунок 2: Результаты работы методов многоклассовой классификации в системе TextonBoost, база MSRC21. Сверху вниз – оригинал изображения, ручная разметка, JointBoost в реализации авторов TextonBoost, AdaBoost.ERP, «один против всех» и собственный улучшенный «один против всех». Области черного цвета (в ручной разметке) не учитываются при подсчете ошибки классификации.

Из результатов видно, что собственный подход на базе метода «один против всех» значительно улучшил работу классического «один против всех». Он стал чаще находить ранее не найденные классы и точнее выделять области. При этом собственный подход иногда работает даже лучше таких сложных методов как JointBoost (в реализации авторов TextonBoost) и adaboost.ERP.

Также целесообразно рассмотреть фактический уровень попиксельной ошибки работы TextonBoost с рассмотренными классификаторами на этих базах (рис. 3).

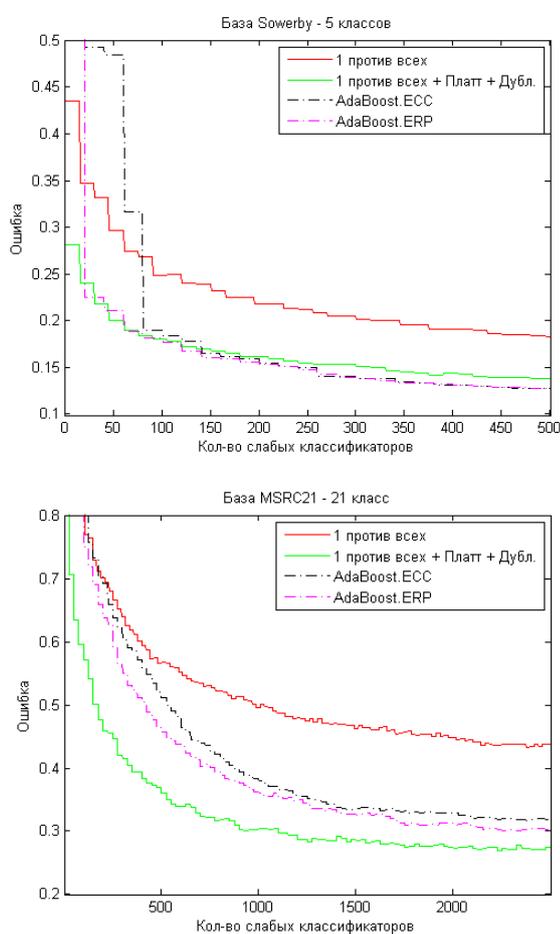


Рисунок 3: Уровень попиксельной ошибки классификаторов на тестовой выборке при работе с TextonBoost на соответствующих базах.

5. ВЫВОДЫ

В результате улучшения метода многоклассовой классификации «один против всех» был получен новый, более точный метод при незначительном изменении алгоритмической сложности классификации (она наименьшая

среди рассмотренных методов). Как показали проведенные эксперименты, он сравним по качеству работы с такими сложными методами, как AdaBoost.ERP и JointBoost.

Вне рамок данной работы было проведено сравнение построенного метода с рассмотренными алгоритмами многоклассовой классификации на базах данных репозитория UCI [12]. В этих экспериментах улучшенный метод «один против всех» также показал сравнимое качество со сложными методами Adaboost.ERP/ERC.

6. ССЫЛКИ

- [1] Jerome Friedman, Trevor Hastie, Robert Tibshirani *Additive logistic regression: a statistical view of boosting* // Annals of Statistics. 2000. Volume 28. № 2. P. 337-407
- [2] Ryan Rifkin, Aldebaro Klautau *In Defense of One-Vs-All Classification* // The Journal of Machine Learning Research. 2004. Volume 4. P. 101-141
- [3] Venkatesan Guruswami, Amit Sahai *Multiclass learning, boosting, and error-correcting codes* // Proceedings of the twelfth annual conference on Computational learning theory. 1999. P. 145-155
- [4] Ling Li *Multiclass boosting with repartitioning* // Proceedings of the 23rd international conference on Machine learning. 2006. P. 569-576
- [5] Chung-Chih Lin, Yuh-Show Tsai, Yu-Shi Lin, Tai-Yu Chiu, Chia-Cheng Hsiung, May-I. Lee, Jeremy C. Simpson, Chun-Nan Hsu *Boosting multiclass learning with repeating codes and weak detectors for protein subcellular localization* // Bioinformatics. 2007. Volume 23. № 24. P. 3374-3381
- [6] Alexandru Niculescu-mizil, Rich Caruana *Obtaining Calibrated Probabilities from Boosting* // In proceedings of the 21st Conference on Uncertainty in Artificial Intelligence. 2005
- [7] А. Кузьмишкина, О. Баринава, А. Коушин *Выбор объектов для обучения в условиях сильной несбалансированности классов* // Труды конференции Graphicon'2008. 2008. стр. 213-216
- [8] Jason Van Hulse, Taghi M. Khoshgoftaar, Amri Napolitano *Experimental perspectives on learning from imbalanced data* // Proceedings of the 24th international conference on Machine learning. 2007. Volume 227. P. 935 – 942
- [9] J. Shotton, J. Winn, C. Rother, A. Criminisi *Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation* // Proceedings of European Conference of Computer Vision. 2006. Volume 3951. P. 1-15
- [10] D. Collins, W. A. Wright *The Sowerby Image Database* // Seventh International Conference on Image Processing And Its Applications. 1999. Volume 1. P. 306 - 310
- [11] MSR Cambridge *Pixel-wise labelled image database* [HTML] (http://research.microsoft.com/en-us/um/people/antcrim/data_objrec/msrc_objcategoimagedatabase_v2.zip)
- [12] UCI Machine Learning Repository [HTML] (<http://archive.ics.uci.edu/ml/>)

About the authors

Sobolev Alexander has graduated from Lomonosov Moscow State University in 2009; Barinova Olga is a junior researcher in Graphics&Media Lab at Lomonosov Moscow State University.