УДК 004.89

Крощенко А.А., Головко В.А., Безобразов С.В., Михно Е.В., Хацкевич М.В., Михняев А.Л., Брич А.Л.

ГЛУБОКОЕ ОБУЧЕНИЕ ДЛЯ ДЕТЕКТИРОВАНИЯ ОБЪЕКТОВ НА ИЗОБРАЖЕНИЯХ ДОКУМЕНТОВ

Введение. Постановка задачи обнаружения объектов. Задача обнаружения объектов на изображениях заключается в выделении отдельных блоков изображения, принадлежащих некоторым заранее определенным классам. Модель, осуществляющая подобную операцию, принимает на вход изображение, а на выходе возвращает координаты и размеры прямоугольных областей, включающих искомые объекты, а также вероятность принадлежности заключенному в них объекту определенному классу.

Решение подобной задачи – актуальная тема в области компьютерного зрения. Фактически благодаря такой функциональности можно осуществлять анализ фото- и видеоизображений в реальном времени, размещая метки на определенных объектах и осуществляя предопределенные операции обработки. При этом нужно отличать задачу обнаружения объектов от задачи семантической сегментации, заключающейся фактически в классификации каждого пикселя изображения. Отличие этих задач проиллюстрировано на рисунке 1.

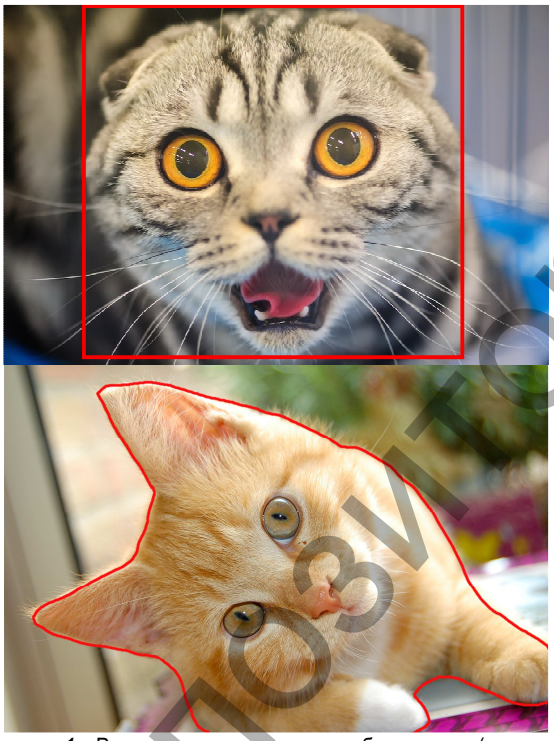


Рисунок 1 – Результат решения задачи обнаружения (детекции) объектов (сверху) и семантической сегментации (снизу)

Задачу обнаружения объекта можно логически разделить на две подзадачи – локализация объекта и его классификация. Многие существующие в настоящий момент подходы к обнаружению объектов на изображениях позволяют объединить эти два разрозненных этапа в одной глубокой нейронной сети, которая выполняет обе задачи одновременно и формирует итоговый результат на выходе (рисунок 2, см. стр. 3). Это позволяет существенно быстрее решить задачу и получить результат, учитывающий сразу все найденные объекты, без необходимости их последовательной обработки [1–5]. Однако не следует в полной мере отказываться от классических подходов – существуют задачи, в которых такие методы дают лучшие результаты.

Оценка качества решения задачи локализации объекта производится вычислением метрики IoU (Intersection over Union), вычисление которой проиллюстрировано на рисунке 3.

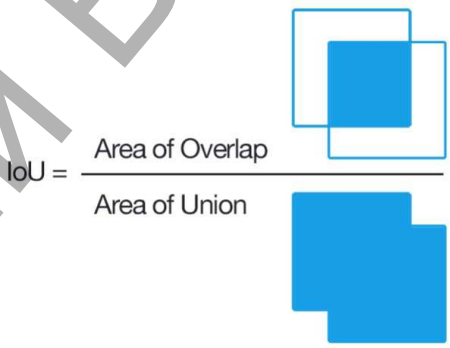


Рисунок 3 – Вычисление метрики IoU

В качестве задачи в данной статье рассматривается задача обнаружения объектов в документах, представленных изображениями. Фактически для такого случая задача обнаружения объектов сводится к задаче разметки электронного документа с выделением его составных частей. Пример анализируемого документа из выборки Doxima7000, предоставленной компанией CIB Software [6], представлен на рисунке 4.

Из представленного изображения видно, что документ состоит из определенных логических блоков (в частности логотип компании, таблица, текст, банковские данные, адрес и т. д.), которые могут быть обнаружены и подвергнуты дальнейшей обработке. Так, например, может быть осуществлено распознавание текстовых блоков с переводом их в формат, который может быть легче проанализирован и интерпретирован с помощью компьютера.

Крощенко Александр Александрович, *ст. преподаватель кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*
Головко Владимир Адамович*, д.т.н., профессор, зав. кафедрой интеллектуальных информационных технологий Брестского государственного технического университета.*
Безобразов Сергей Валерьевич, *к.т.н., доцент кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*
Михно Егор Владимирович, *аспирант кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*
Хацкевич Мария Викторовна, *ст. преподаватель кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*
Михняев Александр Леонидович, *ст. преподаватель кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*
Брич Александр Леонидович, *ст. преподаватель кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*
Беларусь, БрГТУ, 224017, г. Брест, ул. Московская, 267.

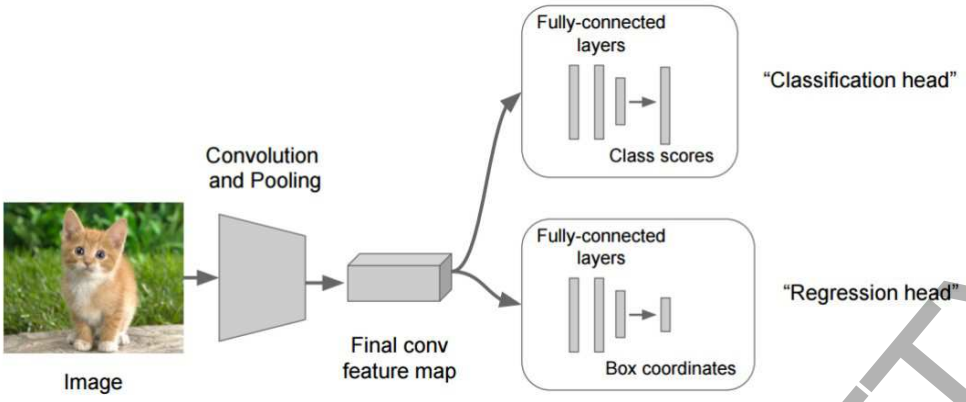
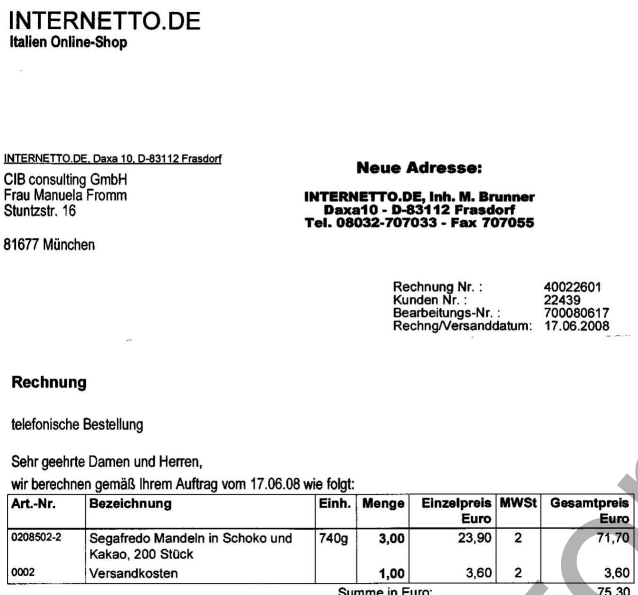


Рисунок 2 – Общий вид нейросетевой модели, применяемой для решения задачи обнаружения объектов на изображении [1]



INTERNETTO.DE
Italien Online-Shop

INTERNETTO.DE, Daxa 10, D-83112 Frasdorf
CIB consulting GmbH
Frau Manuela Fromm
Stuntzstr. 16

81677 München

Neue Adresse:

INTERNETTO.DE, Inh. M. Brunner
Daxa10 - D-83112 Frasdorf
Tel. 08032-707033 - Fax 707055

Rechnung Nr. : 40022601
Kunden Nr. : 22439
Bearbeitungs-Nr. : 700080617
Rechng/Versanddatum: 17.06.2008

Rechnung

telefonische Bestellung

Sehr geehrte Damen und Herren,
wir berechnen gemäß Ihrem Auftrag vom 17.06.08 wie folgt:

Art.-Nr.	Bezeichnung	Einh.	Menge	Einzelpreis Euro	MWSt	Gesamtpreis Euro
0208502-2	Segafredo Mandeln in Schoko und Kakao, 200 Stück	740g	3,00	23,90	2	71,70
0002	Versandkosten		1,00	3,60	2	3,60
				Summe in Euro:		75,30

Рисунок 4 – Фрагмент документа из выборки Doxima7000

1. Обзор применяемых методов для решения задачи обнаружения объектов. Для детектирования объектов на изображениях используется глубокое обучение, которое включает в себя глубокие нейронные сети и их методы обучения [7–18]. В качестве глубоких нейронных сетей для решения задачи обнаружения объектов используются различные варианты сверточных нейронных сетей (CNN):

1. Методы с выделением кандидатов (R-CNN, Fast R-CNN, Faster R-CNN).

2. Однопроходные методы (one-look), к которым относятся SSD, YOLO, YOLO9000.

Основное отличие первой категории от второй – то, что для методов первой категории процесс обнаружения объектов делится на два четких этапа: 1. Локализация регионов-претендентов. 2. Классификация обнаруженных регионов.

Предполагается, что при осуществлении локализации может быть выделено большое количество регионов, не все из которых содержат искомые объекты. В этом случае осуществляется отсев таких регионов, который может быть выполнен как классифицирующей моделью, так и согласно каким-либо другим предположениям.

Рассмотрим применение представленных методов для решения задачи автоматической разметки документов.

1.1. R-CNN (Region-based Convolutional Neural Network). R-CNN базируется на идее метода с предварительным выделением претендентов [2]. Фактически вначале осуществляется анализ изображения с локализацией регионов, предположительно являющихся объектами (всего выделяется около 2000 прямоугольных областей). Принцип работы метода проиллюстрирован на рисунке 5.

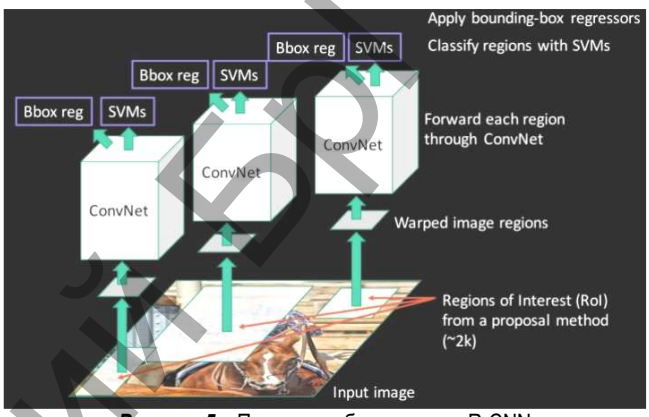


Рисунок 5 – Принцип работы метода R-CNN

Порядок обучения модели R-CNN:

1. Получить какую-либо предобученную сеть (AlexNet, VGG и др.).

2. Переобучить последний полносвязный слой на распознавание объектов, которые должны быть обнаружены.

3. Получить регионы-претенденты (обычно не более 2000) каким-либо способом, масштабировать их до определенного размера (для подачи на CNN), сохранить.

4. Удалить полносвязные слои из базовой CNN.

5. Обучить классификатор на основе SVM для идентификации класса объекта.

6. Обучить линейный регрессионный классификатор для корректировки границ (на выходе получаем смещения dx, dy, dw, dh).

Приведем перечень достоинств и недостатков модели R-CNN.

Достоинства модели:

1. В качестве базовой сети для выделения признаков можно использовать предобученную сеть (например, AlexNet).

2. Метод достаточно интуитивен.

3. Работает быстрее, чем метод скользящего окна.

Недостатки:

1. Скорость обучения и применение подобной модели может быть неудовлетворительным.

2. Общий недостаток всех region proposals-методов – необходимость независимого применения метода выделения регионов-претендентов.

Для решения поставленной задачи была сформирована обучающая выборка. В нее вошло 140 документов, включающих только визитные карточки. Основные классы, которые использовались в качестве целевых для обучения следующие: логотип, отдел, имя, профессия, адрес, телефон, веб-сайт.

Результаты применения метода R-CNN продемонстрированы на рисунке 6.

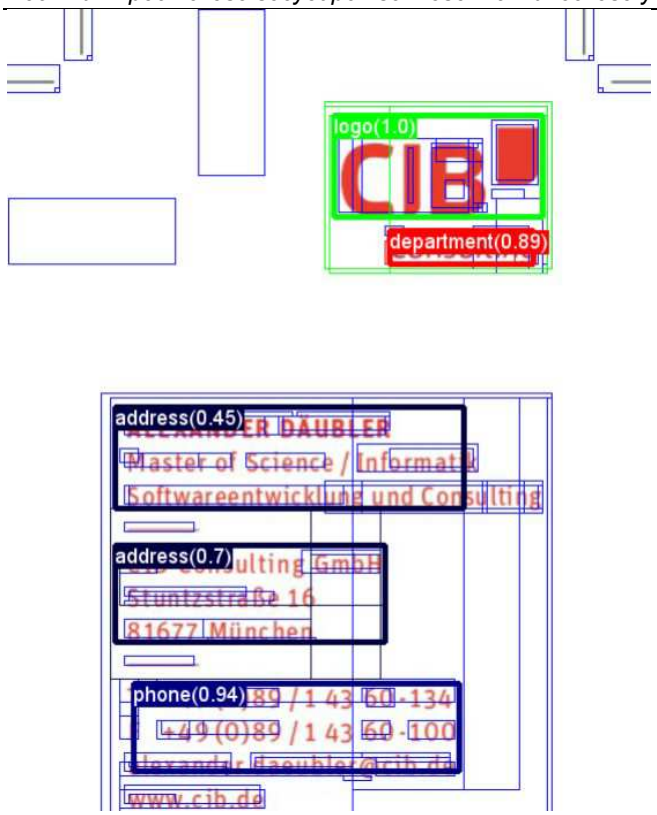


Рисунок 6 – Результат применения метода R-CNN

Ключевой проблемой применения метода R-CNN для решения поставленной задачи стало то, что метод выделяет большое количество претендентов, что кардинальным образом сказывается на скорости его работы. Помимо этого, точность локализации сегментов во многих случаях оказывается неудовлетворительной.

1.2. Fast R-CNN. Метод Fast R-CNN представляет собой развитие идей классического метода R-CNN. В этой архитектуре регионы-претенденты проходят через т.н. слой region of interest pooling (ROI), после чего формируется набор карт признаков фиксированного размера, которые затем подаются на полносвязный слой. Слой ROI pooling фактически действует по принципу подвыборочных (pooling) слоев, отображая проекции интересующих нас областей в области фиксированного (меньшего) размера. Подобная архитектура позволяет уменьшить количество генерируемых блоков и, как следствие, ускорить работу сети. Принцип работы и архитектура модели Fast R-CNN представлены на рисунке 7.

Применительно к поставленной задаче метод показал лучшие по скорости выполнения показатели по сравнению с классическим R-CNN, но качество локализации и распознавания по-прежнему осталось неприемлемым.

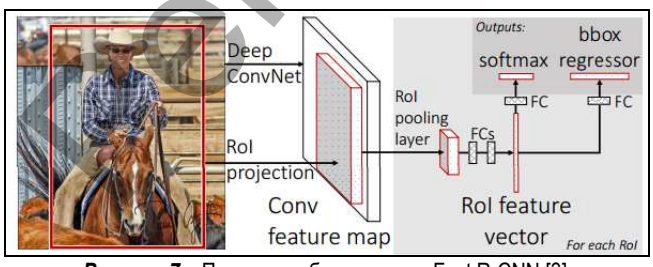


Рисунок 7 – Принцип работы метода Fast R-CNN [3]

1.3. Yolo (You-only-look-once). Сеть Yolo представляет собой первую представительницу группы моделей, позволяющих осуществлять локализацию и распознавание в составе одной единственной модели. Эта архитектура чаще всего базируется на предобученной сверточной нейронной сети (например, VGG16 [4] – рисунок 8), которая используется в качестве составной части модели.

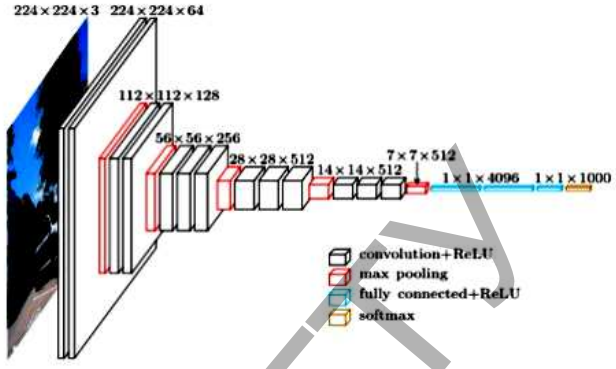


Рисунок 8 – Сверточная сеть VGG16

Предобученная сверточная сеть интегрируется в модель (без своих классифицирующих слоев) и дообучается задачам локализации и распознавания. При этом по сути классы, которые использовались для обучения сверточной нейронной сети, и классы, объекты которых нужно обнаруживать средствами Yolo-сети, могут не совпадать. Фактически предобученная сеть позволяет начать дообучение Yolo-сети с меньшей ошибкой. Архитектура Yolo продемонстрирована на рисунке 9.

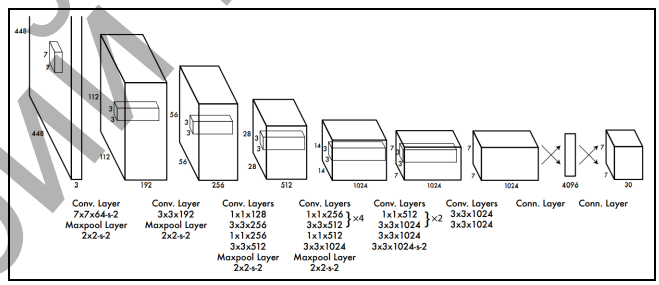


Рисунок 9 – Архитектура Yolo [5]

Алгоритм Yolo осуществляет отображение исходного изображения на решетку размера S x S, где S – размерность карты признаков, и для каждой ячейки этой решетки осуществляет прогнозирование параметров B элементарных прямоугольных областей (регионов, боксов), степень доверия к этим областям (confidence) и метки класса C. Таким образом, такие данные могут быть представлены тензором размерности S x S x (B * 5 + C) (рисунок 10).

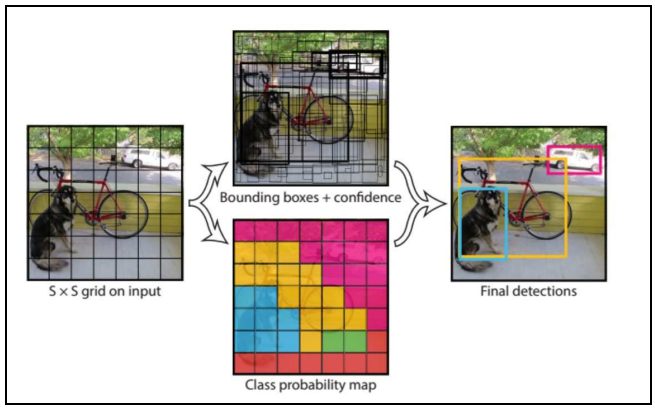


Рисунок 10 – Обнаружение объектов с помощью обученной модели [5]

Приведем порядок обучения модели YOLO.

1. Определение ячейки, близкой к центру «истинного» региона (региона, содержащего объект).

2. Увеличение уровня доверия для регионов, которые перекрываются с истинным регионом с большим значение IoU и уменьшение для тех, которые перекрываются с меньшим.

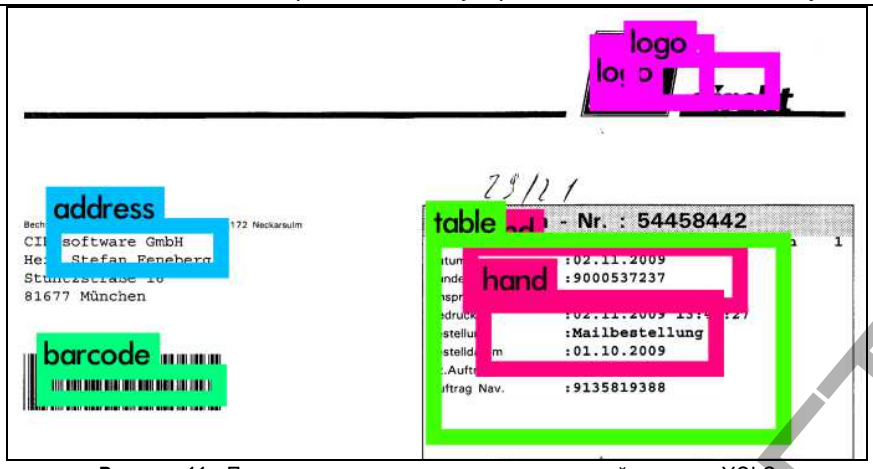


Рисунок 11 – Пример разметки документа, выполненный моделью YOLO

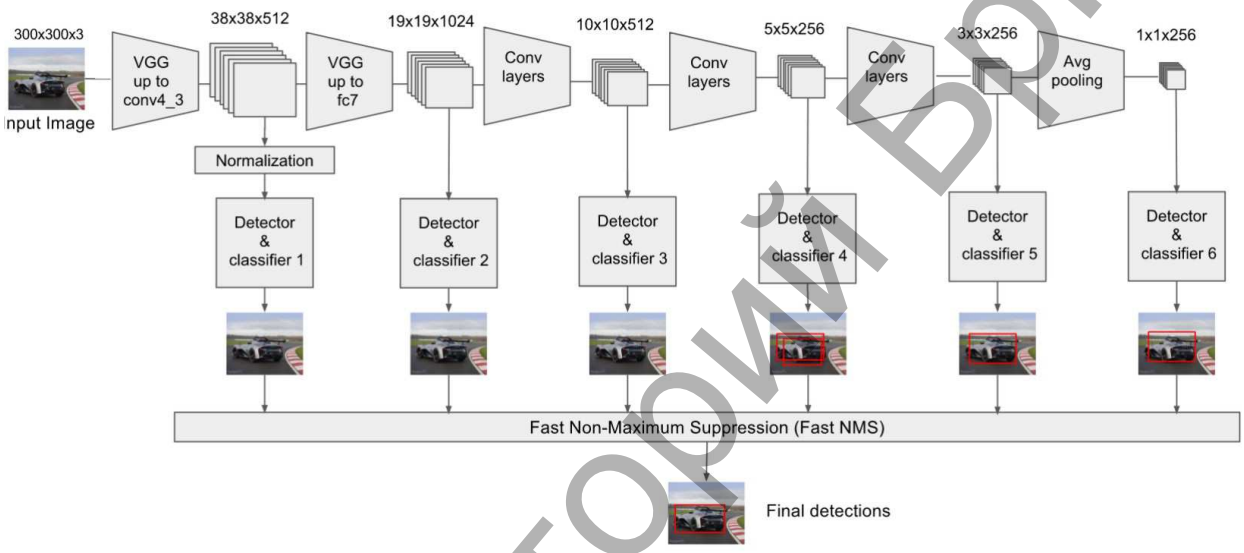


Рисунок 12 – Схема архитектуры SSD

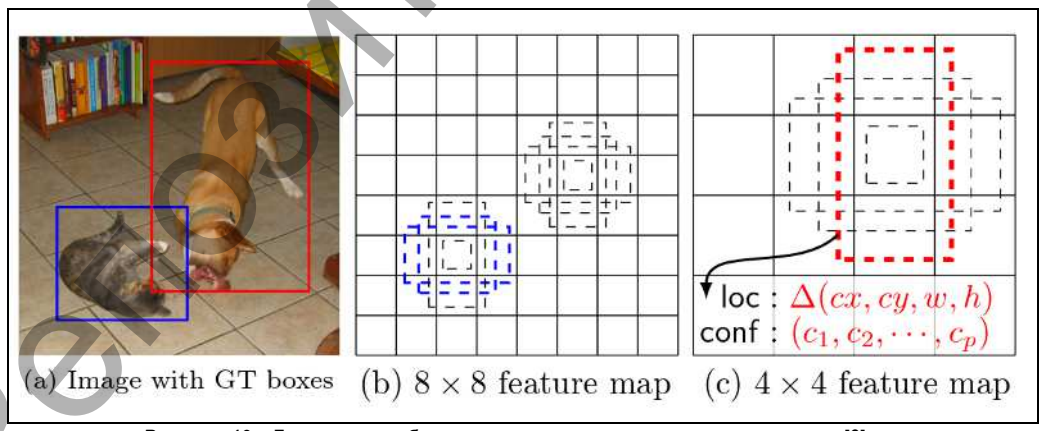


Рисунок 13 – Локализация объектов на картах признаков разных размеров [6]

3. Уменьшение уровня доверия для всех регионов, которые не содержат в себе объекта.

На рисунке 11 проиллюстрировано применение модели YOLO к решаемой задаче.

1.4. SSD (Single-shot detector). Модель SSD [6], как и YOLO, принадлежит к категории однопроходных методов, позволяющих решать задачу обнаружения объектов в рамках одной единственной сети. Схематическое обозначение архитектуры представлено на рисунке 12.

Отметим основные особенности этой модели.

1. Отличается от других singleshot-детекторов (в частности, от YOLO) тем, что каждый слой модели участвует в формировании информации об объектах и их расположении (при этом учитывается масштаб этих объектов - каждый последующий слой обрабатывает объекты большего размера, чем предыдущий) (рисунок 13).

2. В качестве базового элемента используется предобученная сеть (VGG или ResNet), которая преобразуется к полностью сверточной НС (FCN).

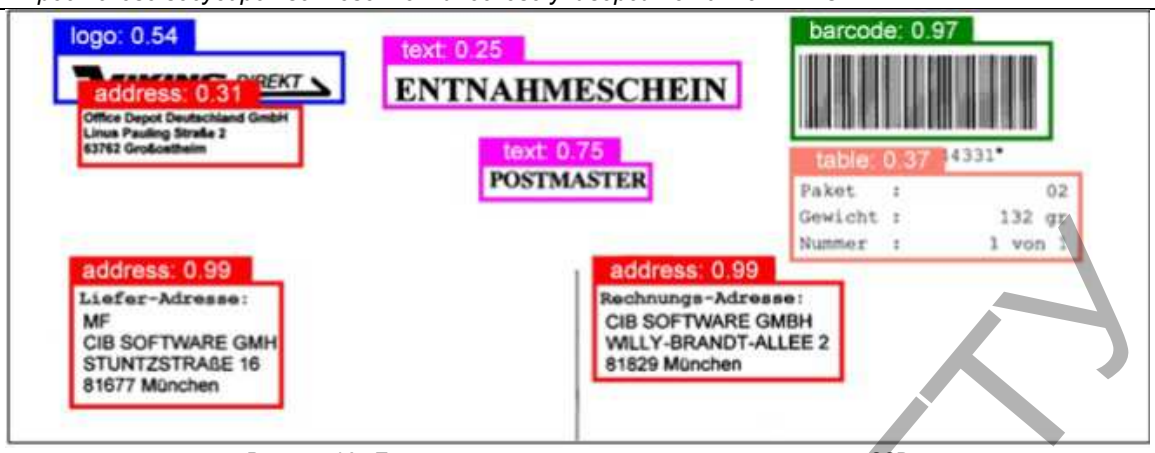


Рисунок 14 – Пример разметки документа, выполненный моделью SSD

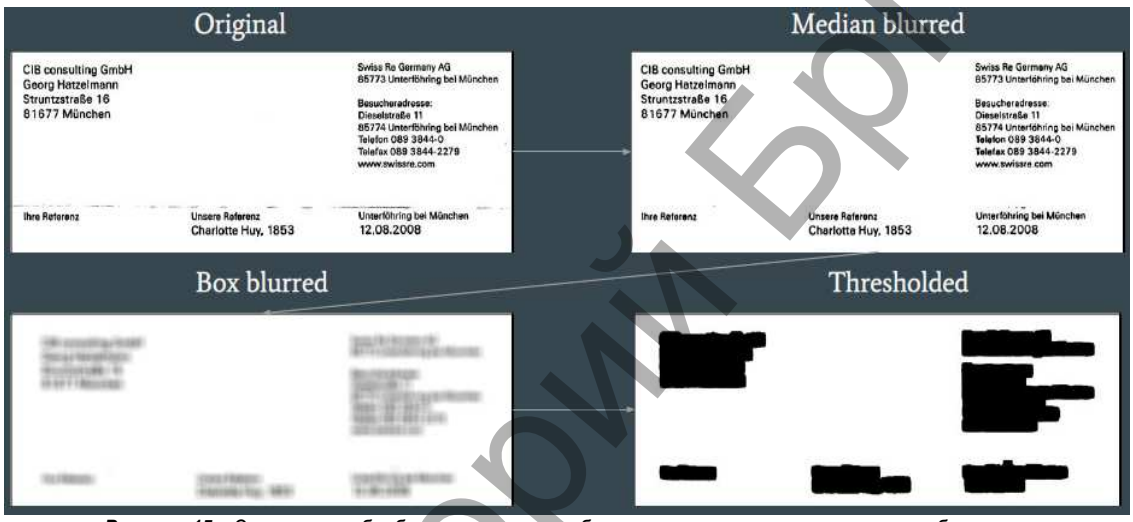


Рисунок 15 – Стадии предобработки исходного изображения для локализации текстовых блоков

3. В процессе работы сети используется Non-maxima suppression для уменьшения количества боксов.

4. Каждый элемент карты признаков формирует набор т.н. default boxes (или Anchors), отличающихся по масштабу и соотношению сторон.

5. Модель обучается, чтобы для каждого anchor правильно прогнозировать его класс и смещение.

Результаты применения архитектуры SSD к решению поставленной задачи показаны на рисунок 14.

2. Построение нейросетевой модели для разметки текстовых изображений. Помимо рассмотрения и изучения стандартных решений для задачи обнаружения объектов, нами был предложен оригинальный подход, основанный на методе R-CNN и включающий в себя два этапа обработки. На первом этапе осуществляется выделение интересующих регионов методов, показавшим преимущество при работе с текстовыми данными. На втором этапе – классификация полученных регионов с помощью классической сверточной сети. Остановимся подробнее на описании каждого этапа обработки.

I. К исходному изображению применяются последовательно следующие операции:

1. Медианный фильтр – для удаления шумов в исходном документе, связанных с неидеальными условиями сканирования документа, его печати и т. д.

2. Бокс-фильтр – линейный фильтр, применяемый для создания эффекта размытия (необходимо для подавления мелких деталей и выделения регионов с однотипным содержимым).

3. Применение пороговой функции для формирования сплошных областей.

4. Выделение контура областей и локализация текстовых блоков.

Первые 3 из указанных операций продемонстрированы на рисунке 15. После выполнения перечисленных операций мы получаем набор прямоугольных областей, содержащих текстовые блоки (рисунок 16).

II. Обучение сверточной нейронной сети. Для распознавания блоков, полученных на первом этапе, обучается сверточная нейронная сеть.

На этом этапе нами была сформирована обучающая выборка, состоящая из порядка 2500 образов, разделенных на 4 класса (логотипы, штрих- и QR- коды и подписи и другое). Нейронная сеть, выбранная в качестве рабочей модели, представлена на рисунок 17.

После выполнения обучения были получены результаты тестирования, представленные в таблице 1. Эти показатели вычислялись следующим образом.

Индекс активируемого нейрона последнего слоя, соответствующий распознанному классу, определялся по формуле:

$$b_s = \underset{i}{\operatorname{argmax}}\, y_i ,$$

где y_i представляет значение i-го нейрона последнего слоя CNN-сети, $i = 1,\ldots,N$, N – количество нейронов в выходном слое, b_s – метка, s = 1, 2, 3,…, L, где L – число изображений в выборке.

Точность рассчитывалась по формулам:

$$A = \frac{S}{L} \times 100\% ;$$

$$S = \sum_{s=1}^{L} 1(b_s = e_s) ,$$

где $1()$ – индикаторная функция, e_s – эталонное значение (метка).

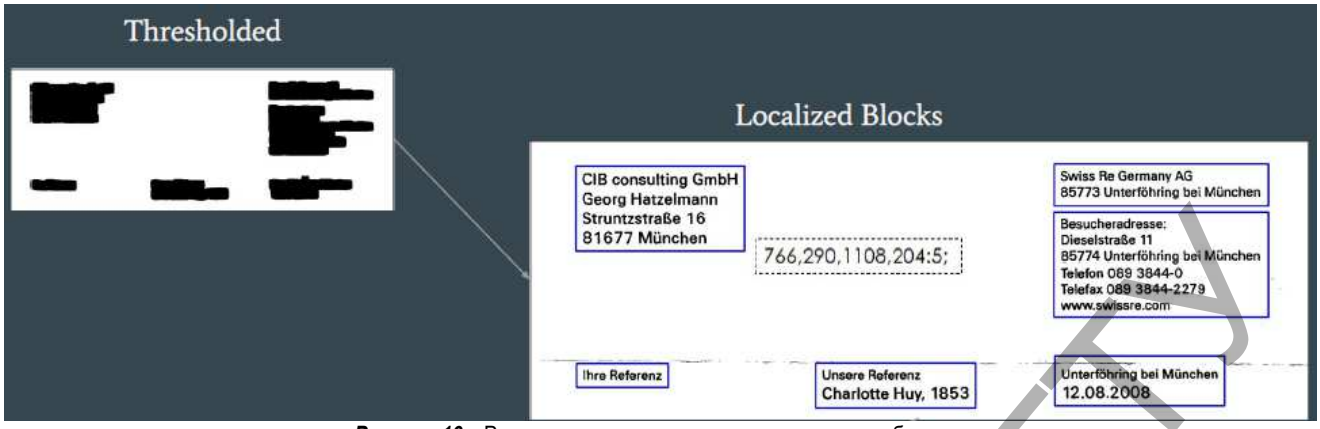


Рисунок 16 – Результат выполнения этапа локализации объектов

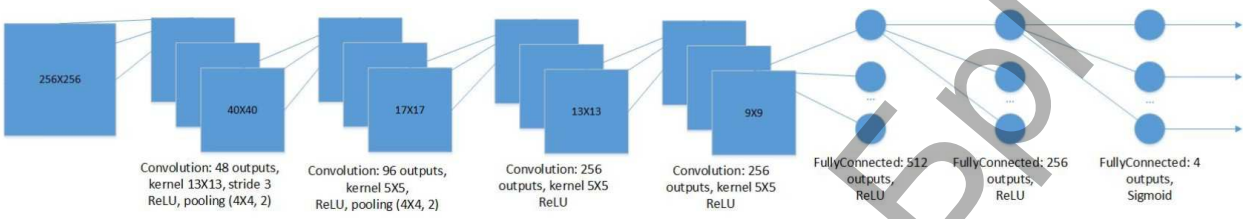


Рисунок 17 – Сверточная нейронная сеть для классификации текстовых блоков

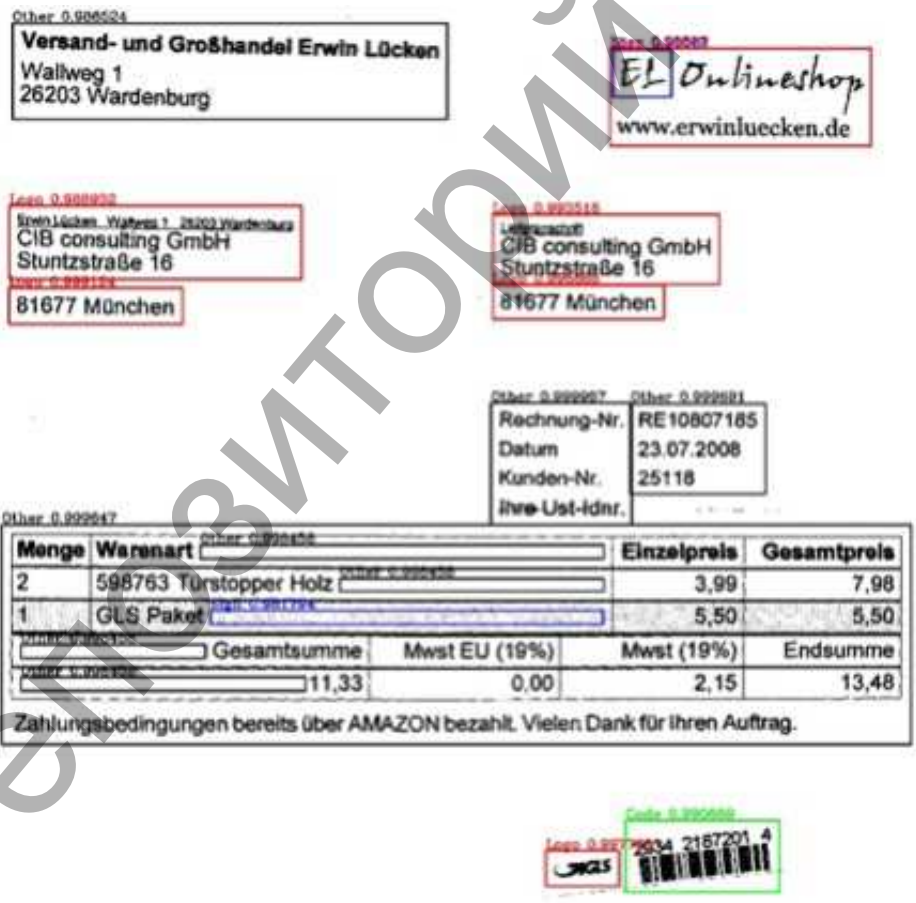


Рисунок 18 – Результат работы системы обнаружения объектов

Таблица 1 – Результаты тестирования

Логотипы	Штрих-коды	Подписи	Другое
97.65	100	97.76	99.29

Таким образом, были получены достаточно высокие показатели эффективности для обученного классификатора.

На рисунке 18 представлен результат работы системы для распознавания некоторого типового документа.

3. Разработка программы полуавтоматической разметки. Формирование обучающей выборки для решения задачи обнаружения объектов является трудоемким и ответственным процессом. В целях упрощения и организации единообразной обработки нами была раз-

работана вспомогательная утилита полуавтоматической разметки, позволившая существенно облегчить и ускорить этап подготовки данных. Рассмотрим ее функциональные возможности подробнее.

1. Возможность работы с распространенными типами изображений.
2. Выделение объектов, принадлежащих разным классам.
3. Пакетная обработка (последовательная загрузка файлов и их разметка) с отображением количества необработанных файлов.
4. Изменение масштаба обрабатываемого изображения.
5. Сохранение в файл результата разметки.
6. Простой формат получающейся разметки.

Семантика формата разметки имеет следующий вид:

регион := <координатаX>,<координатаY>,<длина>,<ширина>:<индекс_класса>

запись := <имя_файла>-регион[;регион]

Таким образом, файл разметки состоит из записей, каждая из которых описывает один из файлов выборки. На рисунке 19 изображено главное окно программы с открытым и размеченным файлом.

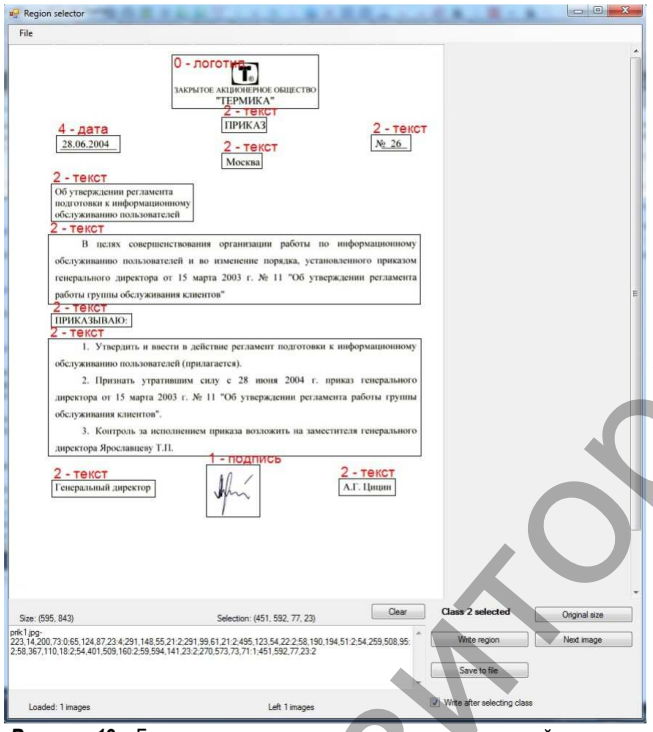


Рисунок 19 – Главное окно программы полуавтоматической разметки

Заключение. В данной статье рассмотрены глубокие сверточные нейронные сети для решения задачи обнаружения объектов на изображении и их классификации. В результате получены следующие результаты:

1. Проведен сравнительный анализ различных методов и архитектур, применяемых для решения задачи обнаружения объектов, который показал, что так называемые однопроходные (one-look) архитектуры нейронных сетей обеспечивают высокое качество решения поставленной задачи.
2. Разработан нейросетевой алгоритм разметки изображений текстовых документов, базирующийся на предобработке изображения, упрощающей локализацию отдельных частей документа и последующее распознавание локализованных блоков с помощью глубокой сверточной нейронной сети. Полученный алгоритм показал высокое качество локализации и приемлемый уровень последующей классификации.
3. Разработана программа полуавтоматической сегментации, позволяющая значительно упростить и упорядочить ручной труд по разметке изображений и подготовке обучающей выборки для решения задач обнаружения и классификации объектов.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Object Localization and detection // leonardoaraujosantos.gitbooks.io [Web-resource]. – 2017. – Mode of access: https://leonardoaraujosantos.gitbooks.io/artificial-inteligence/content/object_localization_and_detection.html. – Date of access: 12.12.2017.
2. Girshick, R. Rich feature hierarchies for accurate object detection and semantic segmentation / R. Girshick, J. Donahue, T. Darrell, J. Malik // arXiv [Web-resource]. – 2014. – Mode of access: https://arxiv.org/pdf/1311.2524v5.pdf. – Date of access: 12.12.2017.
3. Girshick, R. Fast R-CNN / R. Girshick // arXiv [Web-resource]. – 2015. – Mode of access: https://arxiv.org/pdf/1504.08083.pdf. – Date of access: 12.12.2017.
4. Simonyan, K. Very Deep Convolutional Networks for Large-scale Image Recognition / K. Simonyan, A. Zisserman // arXiv [Web-resource]. – 2015. – Mode of access: https://arxiv.org/pdf/1409.1556.pdf. – Date of access: 12.12.2017.
5. Redmon, J. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon, S. Divvala, R. Girshick, A. Farhadi // arXiv [Web-resource]. – 2016. – Mode of access: https://arxiv.org/pdf/1506.02640.pdf. – Date of access: 12.12.2017.
6. Cib software – немецкий стандарт информационных технологий с 1989 года // cib.by [Веб-ресурс]. – 2017. – Режим доступа: https://cib.by/ru. – Дата доступа: 12.12.2017.
7. Hinton, G. Training products of experts by minimizing contrastive divergence / G. Hinton // Neural Computation, 14. – 2002. – P. 1771–1800.
8. Hinton, G. Reducing the dimensionality of data with neural networks / G. Hinton, R. Salakhutdinov // Science, 313 (5786). – 2006. – P. 504–507.
9. Hinton, G. A practical guide to training restricted Boltzmann machines // Tech. Rep. 2010-000. – Toronto: Machine Learning Group, University of Toronto, 2010.
10. Bengio, Y. Learning deep architectures for AI // Foundations and Trends in Machine Learning. – 2009. – Vol. 2(1). – P. 1–127.
11. Bengio, Y. Greedy layer-wise training of deep networks / Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle // In book Schölkopf, J.C. Platt, T. Hoffman (Eds.), Advances in neural information processing systems, 11. – MA: MIT Press, Cambridge, 2007. – P. 153–160.
12. Hinton, G. A fast learning algorithm for deep belief nets / G. Hinton, S. Osindero, Y. Teh // Neural Computation. – 2006. – Vol. 18. – P. 1527–1554.
13. Golovko, V. The Nature of Unsupervised Learning in Deep Neural Networks: A New Understanding and Novel Approach / Vladimir Golovko, Aliaksandr Kroshchanka, Douglas Treadwell // Optical Memory And Neural Networks (Springer Link). – 2016. – Vol. 25, № 3. – P. 127–141.
14. Golovko, V.A. Deep learning: an overview and main paradigm / V.A. Golovko // Optical Memory and Neural Networks (Information Optics). – 2017. – Vol. 26(1). – P. 1–17.
15. Golovko, V. A Learning Technique for Deep Belief Neural Networks / V. Golovko, A. Kroshchanka, U. Rubanau, S. Jankowski // in book Neural Networks and Artificial Intelligence. – Springer, 2014. – Vol. 440. Communication in Computer and Information Science. – P. 136–146.
16. Golovko, V. A New Technique for Restricted Boltzmann Machine Learning / Aliaksandr Kroshchanka, Volodymyr Turchenko, Stanislaw Jankowski, Douglas Treadwell // Proceedings of the 8th IEEE International Conference IDAACS-2015. – Warsaw, 2015. – P. 182–186.
17. Golovko, V. Theoretical Notes on Unsupervised Learning in Deep Neural Networks/V. Golovko, A. Kroschanka//Proceedings of the 8-th International Joint Conference on Computational Intelligence, NCTA'2016, Porto, Portugal, 9-11 November 2016. – P. 91–96.
18. Jankowsky, S. Deep learning classifier based on NPCA and orthogonal feature selection / Stanisław Jankowski, Zbigniew Szymański, Uladzimir Dziomin, Vladimir Golovko, Aleksy Barcz //Proc. SPIE 10031, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2016, 100315E (September 28, 2016); 2016. – SPIE, 2016. – P. 697–702.

Материал поступил в редакцию 13.02.2018

KROSHCHENKO A.A., GOLOVKO V. A., BEZOBRAZOV S.V., MIKHNO E.V., KHATSKEVICH M.V., MIKHNYAEV A.L., BRICH A.L. Deep training for detecting of objects at images of documents

This paper describes deep convolutional neural networks for objects detection and classification. A comparative analysis of various deep techniques and architectures for object detection are carried out. A neural network algorithm for marking up images of text documents was developed, based on preprocessing an image that simplifies the localization of individual parts of the document and subsequent recognition of localized blocks using a deep convolutional neural network. A program of semi-automatic segmentation has been developed that makes it easier to prepare a training data set for object detection and classification.

УДК 004.89

Крощенко А.А., Головко В.А., Безобразов С.В., Михно Е.В., Рубанов В.С., Кривулец И.Ю.

ОРГАНИЗАЦИЯ СЕМАНТИЧЕСКОГО КОДИРОВАНИЯ СЛОВ И ПОИСКОВОЙ СИСТЕМЫ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ

Введение. Задача семантического кодирования приобрела особую важность с развитием поисковых систем. Актуальность подобных технологий связана в первую очередь с возможностью осуществления поиска в больших по объему базах. При этом особое значение имеет не столько нахождение идентичных слов, сколько осуществление поиска близких по некоторой семантической метрике слов.

Интуитивно понятно, что близкие по смыслу слова в предложении должны появляться в одних и тех же или похожих контекстах. Под контекстом в данном случае понимаются слова, располагающиеся в непосредственной близости от рассматриваемого или, иначе, целевого слова. Именно эта идея положена в основу методов семантического кодирования (например, [1, 2]). Эти методы позволяют для словаря *D* фиксированного размера, слова которого представлены в некотором коде, выполнить его преобразование в код меньшей (редуцированной) размерности (рис. 1). Параллельно с этим, благодаря специфике реализации таких методов, происходит выделение семантически значимой информации, которая может быть использована для осуществления функций поиска.

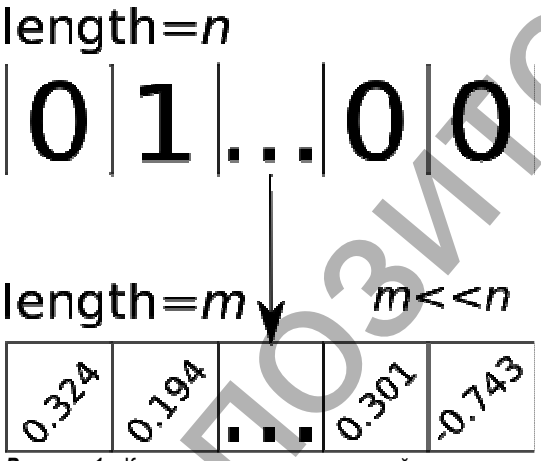


Рисунок 1 – Кодирование слов с редукцией размерности

В силу того, что слова в словаре некоторого языка почти всегда отличаются по длине, реализация какой-либо задачи сравнения слов существенно усложняется. Приведение же каждого слова словаря к вектору заданной размерности, одинакового по длине для всех слов, позволяет осуществлять сравнение искомого и проверяемого слов непосредственно путем вычисления любой (например, евклидовой метрики). Такая технология позволяет не только упростить задачи поиска, но и сделать такой поиск более интеллектуальным.

1. Метод word2vec. Одним из методов семантического кодирования, широко применяемых на практике, является word2vec. Этот подход был предложен Миколовым в 2013 году [1].

Word2vec позволяет осуществлять семантический анализ текста с выделением наиболее близких по смыслу слов. Существует два варианта метода word2vec (рис. 2), отличающихся политикой участия контекста. Под контекстом в данном случае понимается совокупность слов (слева и справа), окружающая целевое слово, взятая в пределах определенного окна.

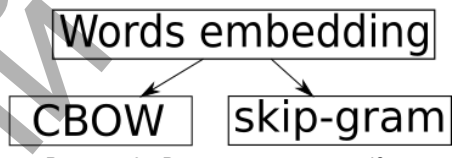


Рисунок 2 – Варианты метода word2vec

Первый вариант, называемый skip-gram, базируется на обучении нейросетевой модели, которая осуществляет формирование контекста на основе одного целевого слова, подаваемого на вход модели (рис. 3).

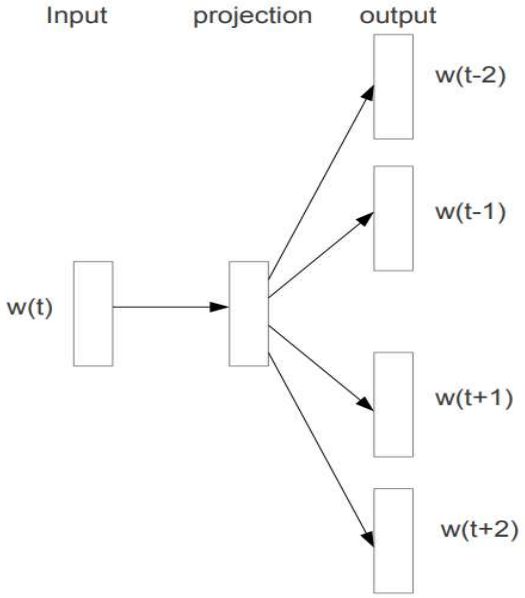


Рисунок 3 – Базовая модель НС метода word2vec (вариант skip-gram) [1]

Рубанов Владимир Степанович, *кандидат физ.-мат. наук, доцент кафедры высшей математики Брестского государственного технического университета.*
Беларусь, БрГТУ, 224017, г. Брест, ул. Московская, 267.
Кривулец Игорь Юрьевич, *аспирант кафедры информационных систем управления Белорусского государственного университета.*
Беларусь, БГУ, 220050, г. Минск, пр. Независимости, 4.