Основными вариантами использования ПК являются:

- загрузка объекта - полупроводниковой пластины или фотошаблона, ориентация в пространстве, перемещение в рабочую зону;
- выгрузка – удаление объекта из рабочей зоны в хранилище (контейнер, кассету);
- инициализация установки и базирование механизмов – загрузка в установку данных, описывающих исходное состояние оборудования для решения конкретной задачи, и установка механизмов в соответствующее состояние/положение;
- управление перемещениями координатного стола – формирование обобщенных команд для управления движением координатного стола;
- совмещение и ориентация – привязка системы отсчета и координатной системы объекта к координатной системе установки;
- контроль и измерение размеров – запуск алгоритмов контроля и измерения размеров;
- автоматическое измерение – запуск алгоритмов автоматического измерения размеров;
- определение размеров элементов – запуск алгоритмов определения размеров изображения;
- управление механизмами – подготовка команд управления оборудованием (метакоманд) и соответствующих параметров;
- формирование управляющих команд – преобразование метакоманд в формат требуемых тем либо иным микроконтроллером управления оборудованием;
- создание программы контроля и измерений для автоматического режима;
- сохранение результатов контроля и измерений – компоновка результатов работы ПК в структуру, предназначенную для дальнейшего хранения в базе данных и последующее сохранение полученного блока данных с использованием СУБД.

ПК совместим с конкурентоспособным прецизионным оборудованием для изготовления высокоточных оригиналов топологий изделий электронной техники выпускаемым ОАО «КБТЭМ-ОМО» и интегрируется в единый технологический цикл [1] для:

- автоматической фотометрии с прецизионной лазерной системой фокусировки;
- контроля критических размеров полупроводниковых пластин;
- контроля совмещаемости слоев полупроводниковых пластин

**Список цитированных источников**

1. Технологические комплексы интегрированных процессов производства изделий электроники / А.П. Достанко, С.М. Аваков, О.А. Агеев, М.П. Батура [и др.]. – Минск: Беларуская Навука. – 251 с.

УДК 004.912

## НЕЙРОСЕТЕВОЙ МЕТОД АВТОМАТИЧЕСКОЙ ИДЕНТИФИКАЦИИ ЯЗЫКА ТЕКСТА

***Байко С. Л.***

*Брестский государственный технический университет, г. Брест, Беларусь*

*Научный руководитель: Крапивин Ю. Б., канд. техн. наук*

Проблема автоматической идентификации языка текста сегодня актуальна как никогда раннее. С развитием сети Интернет и сохраняющейся тенденцией к глобализации возрастает потребность в обеспечении устойчивых процессов коммуникации между

Репозиторий БрГТУ

людьми по всему миру. Это приводит к необходимости создания и активного применения методов и инструментов автоматической обработки информации, представленной в виде текстов на естественных языках (ТЕЯ). Так, например, создаются, активно пополняются и поддерживаются интернет-каталоги, базы данных документов, которые содержат и обрабатывают информацию различной тематической направленности на десятках, а то и сотнях различных языков. А это требует решения задачи автоматической идентификации языка текста документа на самых ранних этапах его обработки. Качество решения указанной задачи во многом определяет результаты последующих этапов.

Анализ существующих методов автоматической идентификации языка текста показал преобладание статистических методов. Большинство популярных решений основано на n-грамм-моделях. Они показывают высокую точность, но являются ресурсоёмкими и требуют больших вычислительных мощностей [1].

Однако прогресс не стоит на месте и в последнее время наблюдается «взрыв» интереса к искусственным нейронным сетям, которые применяются для анализа данных в самых различных областях – медицина, бизнес, техника, физика. Сфера автоматической обработки ТЕЯ не стала исключением – существуют нейросетевые методы решения задачи автоматической идентификации языка текста. Для этого можно применять различные архитектуры нейронных сетей – это и простейшие многослойные персептроны, и более сложные рекуррентные (Recurrent neural networks или RNN) и свёрточные (Convolutional neural networks или CNN) нейронные сети [2]. Одно из основных преимуществ нейросетевого подхода состоит в том, что на этапе обучения искусственные нейронные сети могут распознать более глубокие, иногда неожиданные закономерности в данных, а на этапе анализа – демонстрируют способность к обобщению, т. е. предоставляют возможность получать обоснованный результат на основании данных, которые не встречались в процессе обучения.

Для решения задачи автоматической идентификации языка текста больше всего подходят двунаправленные нейронные сети. «Традиционные» архитектуры искусственных нейронных сетей, такие как, например, многослойный персептрон, имеют фиксированное число входов, и данные с каждого из них воспринимаются независимо. Это является главным недостатком при решении задач в области автоматической обработки ТЕЯ, т. к. входные последовательности – это тексты на ЕЯ, которые имеют смысловую целостность и связность. Рекуррентные нейронные сети решают эту проблему. Архитектура RNN предполагает возможность обмена информацией между искусственными нейронами: например, вдобавок к новому входному массиву данных на вход подаётся предыдущее состояние нейронной сети. Так, реализуется память. Двунаправленная сеть, кроме предыдущих элементов последовательности данных, и учитывает также и будущие.

Выбранная архитектура искусственной нейронной сети состоит из следующего набора слоёв: слой вложений, слой предыдущего состояния и слой будущего состояния. Слой вложений служит для преобразования входного числа в вектор. Слои предыдущего и будущего состояния представляют собой наборы ячеек управления памятью, которые позволяют учитывать состояние последовательности. Входная последовательность – это ТЕЯ. Каждый символ текста подаётся на входной слой нейронной сети – слой вложений. Выходным значением нейронной сети является язык, на котором написан сим-

Репозиторий БРГТУ

вол. В качестве ячеек управления памятью могут применяться LSTM (долгосрочная кратковременная память) и GRU (управляемые рекуррентные блоки).

Полученная архитектура нейронной сети обучается с учителем, из чего следует, что для её обучения необходим набор данных. Таким набором стал Wikipedia Language Identification Dataset [3].

Для того чтобы кодировка не влияла на результаты работы нейронной сети, производится unicode-нормализация, позволяющая привести символы к определенному стандартному виду.

Для обучения и тестирования нейронной сети было разработано клиент-серверное приложение.

Тестирование различных конфигураций нейронной сети показало, что использование ячеек управления памятью типа GRU является наиболее эффективным: сеть с такими ячейками за 45 часов обучения достигла точности идентификации языка в 86,79%. К тому же она использует меньше памяти.

Сравнение работы полученной обученной модели искусственной нейронной сети с наиболее популярными программами идентификации языка текста, такими как LangDetect, LangID и CLD2, основанными на использовании n-грамм-моделей, показало преимущества выбранного метода решения поставленной задачи (таблица 1).

Таблица 1 – Результаты тестирования

| **Система** | **Количество языков** | **Метрика** | | |
|---|---|---|---|---|
| | | **Точность** | **Полнота** | **$F_1$-мера** |
| LangDetect | 50 | 0,974 | 0,982 | 0,978 |
| LangID | 50 | 0,974 | 0,979 | 0,977 |
| CLD2 | 50 | 0,974 | 0,983 | 0,978 |
| Нейросетевой метод | 50 | 0,977 | 0,980 | 0,979 |

В качестве метрик для сравнения были выбраны точность, полнота и $F_1$-мера. Точность характеризует, сколько полученных от классификатора ответов являются правильными. Полнота характеризует способность классификатора находить правильные ответы. $F_1$-мера объединяет в себе информацию о точности и полноте.

Таким образом, можно сделать вывод, что реализованный метод идентификации языка работает не хуже остальных, при этом обладая рядом преимуществ:

1.) большее количество поддерживаемых языков;

2.) лёгкая масштабируемость;

3.) гибкая конфигурация.

Результаты проведённых исследований внедрены в Учреждении образования «Брестский государственный технический университет».

**Список цитированных источников**

1. Крапивин, Ю.Б. Автоматическое определение языка текстового документа для основных европейских языков / Ю.Б. Крапивин // Информатика. – 2011. - №3 (31). – С. 112-117.

2. Jauhiainen, T. Automatic Language Identification in Texts: A Survey / T. Jauhiainen, M. Lui, M. Zampieri, T. Baldwin, K. Linden // Journal of Artificial Intelligence Research. – 2018. – 104 p.

3. Thoma, M. The WiLI benchmark dataset for written language identification. [Электронный ресурс] – Режим доступа: URL: https://arxiv.org/abs/1801.07779 – Дата доступа: 23.10.2019.

Репозиторий БрГТУ