



Рисунок 1 – Модель искусственного нейрона.

Математически выход Y нейрона можно записать в виде следующей формулы:

$$Y = f(S) = f\left(\sum_{i=1}^n w_i x_i + w_0\right) \quad (1)$$

Для целей прогнозирования погодных условий целесообразно применить схему многослойного персептрона с обратным распространением ошибки. Многослойный персептрон позволяет аппроксимировать любую непрерывную функцию или границу между классами со сколь угодно высокой точностью. В качестве функции активации f используется сигмоидная функция (2).

$$f(x) = \frac{1}{1+e^{-x}} \quad (2)$$

Для обучения сети используется алгоритм обратного распространения ошибки:

- 1) инициализировать синаптические веса маленькими случайными значениями;
- 2) выбрать очередную пару из обучающего множества; подать входной вектор на вход сети;
- 3) вычислить выход сети;
- 4) вычислить разность между выходом сети и требуемым выходом;
- 5) подкорректировать веса сети для минимизации ошибки;
- 6) повторять шаги с 2 по 5 для каждого вектора обучающего множества до тех пор, пока ошибка на всем множестве не достигнет приемлемого уровня.

Данная модель позволяет достаточно точно предсказывать численные характеристики погоды с высокой точностью. Однако, такой метод не подходит, если входные данные характеризуются высоким шумом или большим пропуском значений во временном ряде.

Список использованных источников:

1. Bregman, J.I. Environmental Impact Statements / Bregman, J.I. – Boca Raton : MI Lewis Publication, 2006. – 264 p.
2. Folorunsho O. Application of Data Mining Techniques in Weather Prediction and Climate Change Studies / Folorunsho O. // I.J. Information Engineering and Electronic Business. – 2012. – p. 50-51.

РАЗРАБОТКА ПРОГРАММНОГО СРЕДСТВА ГЕНЕРАЦИИ UML- ДИАГРАММ НА ОСНОВЕ ОПИСАНИЯ НА МЕТАЯЗЫКЕ

Кобец И. А.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Смолякова О.Г. – к.т.н., доцент

В статье рассматриваются вопросы разработки программного средства для визуализации UML-диаграмм на основе описания на проблемно-ориентированном метаязыке. Приводится общее описание языка и составляющих его компонентов: отвечающих за описание генерируемых диаграмм и их характеристик, а также используемых для расширения языка

Разработанное программное средство представляет собой транслятор кода на метаязыке в векторное либо растровое изображение UML-диаграммы. Необходимость описания диаграмм на метаязыке может возникать при использовании систем компьютерной вёрстки наподобие Latex, либо разнообразных облегчённых языков разметки, таких как Markdown. Одним из преимуществ автоматической генерации диаграмм на основе описания на метаязыке, является то, что увеличивается скорость их получения, так как нет необходимости в ручном размещении элементов диаграммы. Такая необходимость может появляться только в редких случаях, когда необходимо определённое расположение элементов, для большей выразительности получаемого изображения. Как показала практика проекта PlantUML, текстовое описание UML-диаграмм позволяет слабовидящим студентам и инженерам как разрабатывать, так и читать UML-диаграммы [1,2].

Существует несколько подобных программных средств, самые популярные среди которых PlantUML и Graphviz. Среди основных недостатков у всех аналогов можно выделить слишком общий синтаксис для любых видов диаграмм, описываемых при помощи предметно-ориентированных языков данных программ. Из-за того, что текст на общем для всех видов диаграмм языке сложен для синтаксического анализа, то это не позволяет создать удобные средства для произведения статического анализа кода, автодополнения, навигации, поиска, подсветки синтаксиса, форматирования, а также генерации кода.

При работе над языком разрабатываемого программного средства были учтены недостатки аналогов. И было принято решение, что в метаязыке необходимо обеспечить строгую типизацию, которая бы разграничивала как типы диаграмм и их элементы между собой, так и разницу между используемыми графическими нотациями: UML, IDEF0, IDEF1X и так далее. Также, для упрощения разработки необходимых инструментов для написания кода на данном метаязыке, было решено описать язык с использованием грамматики разбирающей выражение, т.к. они являются однозначными, что упрощает написание синтаксического анализатора [3].

В результате разработки метаязыка было решено выделить два его подмножества: язык для описания диаграмм и элементов их составляющих, а также язык для описания и расширения типов диаграмм и графических нотаций. Язык для описания диаграмм представляет собой предметно-ориентированный язык, при помощи которого описываются конкретные диаграммы, изображения которых необходимо получить на выходе работы программы. Основные элементы этого подмножества языка – диаграммы. Описание каждой диаграммы представляет собой блок, содержащий в себе перечисление вершин, составляющих диаграмму, а также связей между вершинами. Каждый блок диаграммы содержит информацию о том, к какому типу диаграмм он относится, что позволяет использовать внутри блока элементы необходимого типа. Например, Actor и Use Case для диаграммы вариантов использования.

Расположение диаграммы на плоскости происходит в соответствии с выбором способа размещения, каждый тип диаграмм имеет список определённых вариантов визуализации диаграммы, один из которых является способом по умолчанию. Типы размещений для большинства диаграмм – это иерархическое, радиальное, прямолинейное. В архитектуре приложения предусмотрено дальнейшее расширение вариантов расположения. Для расширения существующих нотаций и относящихся к ним типов диаграмм используется второе подмножество языка. В данном подмножестве описывается, как будут выглядеть определенные элементы диаграммы и из каких элементов может состоять диаграмма определённого типа.

В результате разработано расширяемое программное средство, автоматизирующее размещение UML-диаграмм на плоскости и генерацию изображений.

Список использованных источников:

1. Langdon, Patrick, et al. Inclusive designing: joining usability, accessibility, and inclusion. Cham New York: Springer, 2014.
2. Miesenberger, Klaus. Computers helping people with special needs: 13th International Conference, ICCHP 2012, Linz, Austria, July 11-13, 2012, Proceedings. Berlin New York: Springer, 2012.
3. Proceedings of the 31st ACM SIGPLAN-SIGACT symposium on Principles of programming languages. New York, NY: ACM, 2004.

ВЕБ-СЕРВИСЫ ВОПРОСОВ И ОТВЕТОВ

Коваленко И.А.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Парамонов А.И. – к.т.н., доцент

Целью проекта является разработка веб-сервиса вопросов и ответов. Приложение позволит пользователю получить ответ на свой вопрос в автоматическом режиме, используя базу уже имеющихся вопросов и ответов.

Все чаще и чаще люди обращаются за ответами к ресурсам сети Интернет. И если одни пытаются счастья в поисковых системах, то другая половина уже является постоянным пользователем одной из информационных систем вопросов и ответов. Эти системы предоставляют возможность сообществу обмениваться знаниями, зачастую в определенной предметной области.

Ежедневно на подобных сервисах пользователи задают десятки тысяч повторяющихся вопросов. Существующие аналогичные сервисы не анализируют вопрос пользователя перед его публикацией. Соответственно, в таких сервисах, как toster.ru [1], [Ответы @Mail.ru](http://Answers@mail.ru) [2] довольно много повторяющихся пар ответов/вопросов.