

Але існують і недоліки, і найбільшим з них є несанкціоноване копіювання електронної інформації. Щоб запобігти цього, необхідно використовувати локальні сервера, шифрування і обмеження доступу. Особливо важливі документи потрібно обов'язково мати в паперовому варіанті [3]. Також при електронному документообігу можливі збої системи, які можуть привести до безповоротної втрати файлів або їх пошкодження. Щоб не втратити дані потрібно використовувати зовнішню систему резервного копіювання [3].

Отже, електронні документи та документообіг мають свої переваги і недоліки. При переході на електронний документообіг необхідний перегляд поточних процесів і операцій, які в організації залежать від паперового документообігу. Перехід на електронний документообіг неможливо здійснити одноразово, зазвичай це тривалий за часом процес. Потрібно буде врахувати не тільки наслідки відмови від паперових документів, а також затратити час для навчання всіх співробітників користуватися електронним документообігом, тому людський фактор тут особливо важливий, адже від якісної роботи кваліфікованого персоналу буде залежати подальше безперебійне функціонування всієї організації.

Література:

1. Закон України «Про електронні документи та електронний документообіг». Редакція від 30.09.2015, підстава 675-19: (Відповідає офіційному тексту)
2. Електронне урядування: підручник/ В. П. Горбулін, Н. В. Грицяк, А. І. Семенченко, О. В. Карпенко та ін. – К: НАДУ, 2014. — 352 с. – С. 127-129.
3. Гордійчук М. В. Переваги і недоліки електронного документообігу/ М. В. Гордійчук // Матеріали VIII (LX) Міжнародної науково-практичної конференції з філософських, філологічних, юридичних, педагогічних, економічних, психологічних, соціологічних і політичних наук. -2015.

Шаров С.В., к.пед.н., доцент

Печерський Р.В., студент

Мелітопольський державний педагогічний університет

імені Богдана Хмельницького, м. Мелітополь

Кафедра інформатики і кібернетики

АНАЛІЗ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ ЕКСПЕРТНОЇ СИСТЕМИ МЕДИЧНОЇ ЛАБОРАТОРІЇ

Актуальність. Сьогодні ми спостерігаємо повсюдне впровадження інформаційно-комунікаційних технологій в різні сфери діяльності людини, зокрема економіку, науку, освіту. Крім того, багато сучасних програмних засобів використовують елементи штучного елементу, різні

моделі подання знань тощо. Не минули ці процеси і медицину, яка ще з часів розробки експертної системи MYCIN у 1970 році активно використовує інтелектуальні, зокрема експертні, системи по теперішній час. Слід зазначити, що експертні системи призначанні для виконання окремих завдань в окремій наочній області, тому необхідність в медичних експертних систем є актуальною.

Мета статті полягає в аналізі інструментальних засобів для розробки експертної системи медичної лабораторії.

Виклад основного матеріалу. Слід зазначити, що складність процесу розробки експертної системи суттєвим чином залежить від інструментальних засобів, які для цього використовуються. І оскільки на даний момент існує безліч мов та середовищ програмування, призначених для розробки різноманітного програмного забезпечення, при виборі інструментального засобу слід враховувати декілька чинників, а саме:

- специфіки програмного засобу, який буде розроблюватися. Це може бути мобільний додаток, веб-ресурс, віконний додаток та ін.;
- функціональних можливостей, які повинні бути реалізовані.
- досвіду програміста при роботі з конкретним інструментальним засобом;
- часу, який відводиться на розробку.

У своєму прикладному дослідженні ми поставили за мету створити локальну експертну систему, яка повинна працювати на платформі ОС сімейства Windows. Програмний засіб повинен відображати інформацію про види аналізів, обробляти вибрані аналізи у вигляд оформленої заявки від пацієнта, зберігати та аналізувати дані щодо вибраних аналізів. Очікується, що експертна система буде використовуватися співробітником медичної лабораторії, тому повинен бути забезпечений легкий та зручний інтерфейс.

Більшість програмних засобів, які можуть використовуватися для розробки інтелектуальних систем, зокрема експертної системи медичної лабораторії, можна поділити на декілька груп:

1. Мови програмування високого рівня позбавляють розробників від необхідності заглиблюватися в механізми реалізації програмного засобу на машинно-залежному рівні. У цю групу входить дві велика підгрупи: традиційні процедурні мови типу Pascal, C++ та традиційні середовища програмування, які дозволяють створювати додатки з використанням об'єктно-орієнтованого підходу та технології візуального програмування, такі як C# та Delphi.

2. Оболонки експертних систем, які дозволяють створити експертну систему на основі введених правил, параметрів, вбудованого програмного коду. Оболонку можна представити у вигляді експертної систему, яка

містить інструменти та компоненти, які мають загальну спеціалізацію. Прикладом є експертна система EMYCIN, оболонки EXSYS, ЕКО та інші, які дозволяють створювати різноманітні прикладні системи [1, с. 15]. Часто в якості засобів для розробки експертних систем використовують байєсовські мережі довіри (БСД), які використовуються в областях, характеризуються успадкованою невизначеністю. Для байєсовских мереж довіри іноді використовується ще одна назва причинно-наслідковий мережу, в яких випадкові події з'єднані причинно-наслідковими зв'язками [3, с.119]. Оболонки експертних систем вважають досить швидко створити експертну систему, наповнити її знаннями, створюючи таким чином достатньо-функціональний програмний засіб [5, с. 108].

3. Спеціалізовані мови логічного програмування, такі як LISP, Prolog, які застосовуються для створення баз знань та обробки об'єктів та відносин між ними. Для мови логічного програмування Prolog типовими є завдання щодо розробки різноманітних програмних комплексів з елементами штучного інтелекту (експертні системи, комп'ютерні перекладачі, інтелектуальні ігри тощо). Prolog виконує пошук рішення серед опису об'єктів та правил у базі знань за допомогою механізму уніфікації та механізму повторення [4, с. 4].

Для розробки власної експертної системи ми вирішили використовувати мову програмування C# та інтегроване середовище Visual Studio. C# використовується для створення додатків, що працюють на платформі Microsoft .NET Framework. В свою чергу, Visual Studio можна використовувати для розробки різних типів додатків, від самих простих до складних інформаційних систем та веб-порталів. За замовчуванням Visual Studio забезпечує підтримку C#, C, C++, JavaScript, F# і Visual Basic [2].

Висновки. Отже, одним із потужних програмних засобів для розробки експертної системи медичної лабораторії ми вважаємо мову програмування C# та інтегроване середовище Visual Studio, що дозволить створити локальний Desktop-додаток для платформи .Net.

Література

1. Гаврилова Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский. – СПб: Питер, 2000. – 384 с.
2. Интегрированная среда разработки Visual Studio: [Електронний ресурс]. – Режим доступу: [https://msdn.microsoft.com/ru-ru/library/dn762121\(v=vs.140\).aspx](https://msdn.microsoft.com/ru-ru/library/dn762121(v=vs.140).aspx).
3. Тоискин В.С. Интеллектуальные информационные системы: Учеб. пособ. – Ставрополь: Изд-во СГПИ, 2009. – 181 с.
4. Хабаров С.П. Интеллектуальные информационные системы. PROLOG- язык разработки интеллектуальных и экспертных систем: учебное пособие / С.П.Хабаров. – СПб. СПбГЛТУ, 2013. – 138 с.

5. Шаров С.В. Інструментальні засоби та етапи розробки експертних систем / С.В. Шаров, С.О. Хрустальов // Збірник наукових праць Харківського національного університету Повітряних Сил. – 4(49). – 2016. – С. 105–109.

Юсипенко Я.Л., студент

Одесский национальный университет имени И.И. Мечникова

Институт математики, экономики и механики

Кафедра вычислительной математики

ИССЛЕДОВАНИЕ СТРУКТУРЫ МНОГОМЕРНЫХ ДАННЫХ С ПОМОЩЬЮ ПониЖЕНИЯ РАЗМЕРНОСТИ И ВИЗУАЛИЗАЦИИ

Процесс нахождения неявных зависимостей и получения новых знаний из набора данных состоит из различных этапов. На любом этапе анализа данных, визуализация данных играет важную роль, так как позволяет увидеть структуру данных, а также обеспечивает исследователя “интуицией” в дальнейшей работе с данными, позволяет своевременно отказаться от неподходящих моделей.

Задача визуализации - частный случай задачи понижения размерности. Линейные методы понижения размерности упускают из вида нелинейные связи, поэтому такие способы, как отбор на основе корреляции или даже часто используемый РСА могут быть слишком простой моделью для анализа. К нелинейным методам снижения размерности относится метод многомерного шкалирования (MDS) [1]. Он сохраняет попарное расстояние между объектами в маломерном пространстве. Но чем больше размерность исходного пространства, тем меньше расстояние между непохожими объектами в маломерном пространстве. Данный недостаток можно преодолеть, если требовать сохранения не попарных расстояний, а отношений попарных расстояний между объектами (алгоритм t-SNE) [2].

Рассмотрен набор данных, размерность исходного признакового пространства которого равна 41. На рис.1 показано взаимное расположение объектов выборки на плоскости в результате работы алгоритма t-SNE. В двумерном пространстве мы сохранили соотношения расстояний между объектами. Так близкие объекты в исходном многомерном пространстве остаются близкими и в двумерном пространстве.