

(вільний/зайнятий) – (червоний квадратний елемент); кнопка з надписом «Хід» – початок руху ліфта.

Для початку руху за видом *inLift* необхідно обрати поверх (и) і натиснути кнопку «Хід». Якщо поверхів обрано декілька, то після натискання кнопки «Хід» ліфт зупиняється на першому з обраних поверхів відповідно до свого напрямку руху; подальший рух буде здійснено тільки після повторного натискання кнопки «Хід». Після відпрацювання завдань індикатор стану ліфта просигналізує, що ліфт вільний.

Для початку руху за видом *outLift* необхідно натиснути відповідну кнопку поверху, виходячи із необхідності напрямку руху – ліфт одразу поїде на заданий поверх.

Таким чином використання в навчальному процесі наведеного прикладу сприяє набуттю у студентів навичок роботи з будь-якими логічними контролерами, які підтримують роботу з інструментальним програмним комплексом промислової автоматизації *CoDeSys*.

Список використаних джерел:

1. Интерфейс RS-485: описание, подключение. – Режим доступа: http://www.novosoft.by/?page=b_rs_485
2. Описание протокола Modbus. – Режим доступа: <http://onitex.ru/modbus-protokol>
3. Danilevko O. Experience in the developing of the laboratory stand for research of the pump operating modes in Kryvyi Rih national university / O. Danilevko, S. Bondarevskiy // Metallurgical and Mining Industry. – 2015. – № 6. – P. 84-88.
4. CoDeSys – Режим доступа: www.codesys.com

Деркач А.И.

студент,

Национальный авиационный университет

ПРИМЕНЕНИЕ ВЕКТОРНОЙ МОДЕЛИ В МЕТОДАХ ОБРАБОТКИ ТЕКСТОВОЙ ИНФОРМАЦИИ

Векторная модель в информационном поиске – это представление коллекции документов с помощью векторов, принадлежащих коллекции всего векторного пространства. Представление текстовой информации в виде векторов позволяет легко сравнивать слова, искать похожие, проводить классификацию, кластеризацию и другие манипуляции с входными данными в виде текста.

Существует два основных подхода к оптимизации методов обработки текстовой информации, и вообще к сравнению документов по смыслу. Первый подход основан на ручном присвоении объектам некоторых атрибутов и в дальнейшем обработке именно атрибутов, а не самих объектов. Сюда можно отнести тегирование, ручную каталогизацию и онтологию. Вторым подходом, основан на противоположной идее: преобразование текста в набор чисел и

дальнейшую работу с цифрами. Начало этот подход берёт в работах над методом LSA (Latent Semantic Analysis, неявный семантический анализ). На данный момент Google и некоторые другие поисковые системы используют один из параметров данного метода (а именно индекс $tf * idf$) при ранжировании результатов. Основная идея метода довольно простая: чем чаще два слова встречаются в одних и тех же документах (коллекциях), тем ближе они по смыслу. Исходя из этого, можно ожидать, что слова «структура» и «алгоритм» будут встречаться в одних и тех же документах гораздо чаще, чем, например, «структура» и «книга». Для LSA частота встречаемости в конкретном документе зачастую представляется как раз в виде индекса $tf * idf$, что расшифровывается как «term frequency * inverse document frequency».

Term frequency – частота термина – рассчитывается как количество вхождений конкретного термина в конкретный документ, делённое на общее количество слов в этом документе:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Document frequency – частота документа – это количество документов, в которых данный термин встречается, делённое на общее количество документов. Inverse document frequency, соответственно, это величина, обратная document frequency, то есть $idf = 1/df$. Обычно, чтобы как-то смягчить эффект действия

idf на общий результат, вместо самого значения берётся его логарифм:

$$idf_i = \log \frac{|D|}{|\{d: t_i \in d\}|}$$

Индекс $tf * idf$ уже можно использовать для ранжирования результатов поиска – чем выше индекс, тем больше релевантность данного термина в данном документе. Тем не менее, для более сложных задач, таких как сравнение двух документов, этого всё ещё недостаточно. С математической точки зрения термин можно выразить через содержащие его документа просто сложив вектора этих самых документов. И в данном случае не имеет значения, являются ли вектора документов действительно ортогональным (как в LSA) или почти ортогональными. Таким образом, вектор термина будет равен:

$$t_n = \sum_{R(i)} f_{ni} * d_i$$

где $R(i)$ – область определения для векторов документов, т. е. просто номера используемых документов;

d_i – начальный вектор i -ого документа;

f_{ni} – частота n -ого термина в i -ом документе.

Частота здесь может рассчитываться просто как количество вхождений данного термина в документ. В данном случае нормализация частоты не требуется, поэтому храниться и обрабатываться будут целые числа. Слова сравнивать по смыслу непрактично. Сравнивать нужно тексты, т. е. документы. И на данном этапе нужно выбрать, как выразить документы через

содержащиеся в них слова. Легче всего это сделать так же, как и слова через документы: сложив вектора соответствующих объектов:

$$d_m = \sum_{i=1}^{\text{count}(\text{terms})} t_i$$

где $\text{count}(\text{terms})$ – количество терминов (не обязательно уникальных) в документе;

t_i – i -тый термин в тексте;

d_m – вектор m -ного документа.

Векторное представление документов дает несколько плюсов.

Во-первых, векторы можно сравнивать. Два вектора будут тем ближе, чем меньше угол между ними. Для оценки угла достаточно взять любую монотонно-возрастающую функцию, например, коэффициент Отиаи:

$$\cos(\vec{a}, \vec{b}) = \frac{\vec{a} * \vec{b}}{|\vec{a}| * |\vec{b}|}$$

Произведение векторов показывает, насколько похожи два документа, а произведение их длин выступает в роли фактора нормализации. Если произведение не нормализовать, то, например, два документа по 200 слов со всего 10-ю общими элементами в векторах дадут более высокий результат, чем два документа по 20 слов с 6-ю общими элементами.

Во-вторых, над ними можно проводить те же операции, что и над обычными векторами – сложение, вычитание и умножение на скаляр. Интерпретироваться эти операции будут следующим образом: сложение – это объединение двух текстов, умножение на скаляр – повторение одного и того же текста несколько раз и т. д.

После вычисления весов всех слов в тексте, документ может быть представлен как вектор, в котором каждый компонент соответствует отдельному слову документа. Представление коллекции текстов и запросов в виде векторов, входящих в них слов, и составляет суть векторной модели информационного поиска.

Запрос также может быть представлен как вектор весов слов. Таким образом, теперь соответствие запроса документу измеряется конкретным числом, и все документы могут быть упорядочены в выдаче поисковой системы по этому числу.

К преимуществам векторной модели информационного поиска относится то, что модель предоставляет собой несложный набор функций. При этом конкретный способ вычисления весов слов в документе может меняться в зависимости от решаемой задачи и рабочей коллекции. Недостатком подхода является предположение о независимости слов в тексте, что противоречит тому, что в тексте используется множество связанных по смыслу слов.

Список использованных источников:

1. Turney P.D., and Pantel P. (2010), From frequency to meaning: Vector space models of semantics, Journal of Artificial Intelligence Research (JAIR), 37, 141-188.

2. Барсенгян А.А. Анализ данных и процессов. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2009. – 512 с.: ил.

Диесперов А.В.

студент;

Яшков И.О.

кандидат технических наук, доцент,

Харьковский национальный университет радиоэлектроники

ОСОБЕННОСТИ ИДЕНТИФИКАЦИИ ЛИЧНОСТИ ПРИ ПОМОЩИ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ

Сегодня система контроля востребована в предприятиях и деловых центрах. Она выполняет ряд задач: предотвращает проникновение посторонних лиц на территорию, ведёт их учёт на вход и выход. Преимущество автоматизированного контроля – отсутствие человеческого фактора. В данной работе будут рассматриваться электронные замки с использованием электронных ключей.

Электронный замок – электронное устройство, предназначенное для того, чтобы предотвратить доступ в помещение посторонних лиц, или наоборот, ограничить выход из помещения. Решение о доступе лиц в помещение принимается на основе сигналов от различных датчиков: считывателей магнитных карт, штрих-кодов, датчиков контактной памяти, биометрических датчиков, наборной клавиатуры, дистанционного управления и т.д. Часто является частью сложной электронной системы контроля доступа, иногда неотделим от нее. В качестве исполнительных механизмов используются электромеханические и электромагнитные запорные устройства.

Электронный ключ – портативное устройство для отпирания двери с электронным замком, выполненное в виде магнитной карточки, бесконтактного устройства, действующего на некотором расстоянии, или устройства с электрическим контактом.

Преимущество электронных замков – это возможность предоставления доступа одновременно нескольким личностям, не требует замены в случае утери всех ключей. Рассмотрим типы ключей:

1. Магнитная карта

Магнитная карта – это пластиковая карта, которая соответствует спецификациям ISO, имеет на обратной стороне магнитную полосу с информацией объемом около 100 байт памяти, которая считывается специальным считывающим устройством, и место для подписи. Такие магнитные карточки широко используются во всем мире как банковские кредитные и дебетовые карты. Магнитная полоса может быть изготовлена для различных мощностей магнитного поля, и по этому параметру различают высококоэрцитивную (HiCo) и