УДК 519.713

Б.В.ШАМША, Т.Б. ШАТОВСКАЯ, А.Ю. ГУД

ИНТЕЛЛЕКТУАЛЬНАЯ РЕКРУТИНГОВАЯ СИСТЕМА

Центр Карьера является информационной, аналитической и организационной поддержкой трудоустройства студентов и дипломированных специалистов. Для поддержки всех основных направлений деятельности данного центра была разработана информационная система. Сейчас система укрепляет связи между студентами и компаниями, являясь базой резюме и вакансий. С другой стороны, эта система является виртуальным рекрутером, который учитывает личные возможности и предпочтения студентов, доступные вакансии, направления деятельности компаний, локальную инфраструктуру рынка труда, производственные и технологические тенденции, ведёт учёт спецификаций рабочих позиций, доступных человеческих ресурсов для обеспечения эффективных решений в рамках трудоустройства. Данная статья представляет интеллектуальную менеджментную систему, основанную на методах текстового анализа для поддержки рекрутинговых сервисов.

1. Введение

В наши дни одной из наиболее пугающих социальных проблем в Украине является официальная безработица среди молодёжи. Даже после окончания университета молодые профессионалы очень редко могут найти работу, которая была бы адекватна полученным ими специальностям. В частности, гораздо сложнее занять соответствующую вакансию после окончания университета. Тесное сотрудничество между университетами и предприятиями необходимо для стратегического планирования новых «хай-тек» вакансий в компаниях. Такое сотрудничество выгодно для обеих сторон и позволит студентам получить высокие должности в частном и государственном секторах. Два основных класса сервисов, оказываемых Центром Карьера: помогать дипломированным профессионалам найти подходящую работу и помогать компаниям найти профессионалов на вакантные должности.

Университетский Центр Карьера предоставляет студентам консультационную помощь, беря при этом во внимание их личные возможности и предпочтения, доступную работу, учитывает спецификации рабочих позиций, доступные человеческие ресурсы, файлы соискателей (основываясь на университетских дипломах), файлы университета и компаний, национальные образовательные политики и стандарты, инфраструктуру локального рынка труда, производственные и технологические тенденции. С другой стороны, университетским Центром Карьера ведётся множество исследований рынка труда, таких как анализ трудоустройства студентов через базы практики, анализ занятости специалистов и магистров на предприятиях, еженедельная статистика трудоустройства студентов, анализ тенденций востребованности специальностей. Имеется в виду, что Центр Карьера проводит аналитические исследования и предлагает решения для повышения конкурентоспособности среди студентов. Конечно же, это невозможно без соответствующей современной аналитической информационной системы.

Один из модулей системы – виртуальный консультант по трудоустройству**.**

Обработка резюме является непростой задачей. Каждый из нас – личность, и предоставляет информацию по-своему. Инженер университетского Центра Карьера ежедневно рассматривает и анализирует десятки студенческих резюме, сталкиваясь с самыми различными стилями их написания. Форматирование, шрифты и логическая структура резюме абсолютно произвольны. Более того, некоторые компании имеют свою собственную структуру резюме, а некоторые хотят рассматривать и анализировать стиль и логику резюме студентов, которые не имеют большого опыта в их написании. Системы, позволяющие перемещать на них наиболее рутинные действия инженера по обработке резюме и вакансий компаний, сейчас просто необходимы. Два основных свойства такой системы – это автоматический сбор информации о соискателях и вакансиях компаний, кластеризация резюме и вакансий, автоматическое нахождение наиболее подходящих вакансий соответствующим резюме. Другими словами, эта система должна работать как виртуальный интеллектуальный web-консультант для студентов. Необходимо отметить, что существуют критерии, по которым резюме студента отбирается компаниями, но это субъективный подход.

Сейчас процесс сравнения и отбора более новых резюме из ранее существовавших, классификации резюме осуществляются исключительно вручную. Количество резюме возрастает (вместе с количеством их копий от одного и того же претендента), а допустимое время их обработки сокращается. Это приводит к невозможности обработать весь поток поступающих резюме, если этот процесс осуществляется вручную. Создание интеллектуальной web-системы как виртуального рекрутингового консультанта позволяет решить следующие проблемы:

1. Осуществить анализ структуры и распознавание полей резюме для формализованной репрезентации.
2. Автоматический сбор резюме и вакансий с надёжных web-сайтов и их добавление в базу даннх.
3. Классификация всех резюме и вакансий согласно тематикам.
4. Отсеивание дубликатов.
5. Гибкий поиск согласно пользовательским запросам.
6. Ранжирование резюме и вакансий внутри группы: принимается во внимание наличие иерархии в предметной области, используется матрица навыков и обязанностей.
7. Соответствие вакансий наиболее подходящим резюме студентов.
8. Аннотации к резюме и их группам.

2. Обзор метода

Некоторая часть информации о кандидате вносится вручную, что приводит к её искажению. Однако автоматическое извлечение информации не всегда корректно, поэтому для данной проблемы не используется полностью автоматизированный подход. Более подходящим является автоматизированный подход с ручным распознаванием. В большинстве случаев автоматическая обработка даёт хорошие результаты, но когда система не может корректно определить некоторые части резюме, инженер проводит правку вручную. Таким образом, система получает ещё одну копию обучающего образца, который будет использован на следующей обучающей фазе. Также система должна быть в состоянии проверить последовательность резюме. Например, это используется для проверки пересечений периодов работы в различных местах, сверки навыков, указанных в резюме, с описаниями реальных проектов. Как базовая функция, система должна классифицировать поступившие резюме и вакансии по определённым группам и обновлять базу данных.

Части шаблонов резюме не фиксированы, поэтому в целом мы предполагаем, что кандидат и компании присылают резюме и вакансии в произвольной форме. Резюме соискателя всегда состоит из частей, другими словами, имеет логическую структуру. Логические блоки обычно имеют названия. Это позволяет разместить их в нужных местах с помощью методов текстовой кластеризации [1-4]. Несмотря на факт, что стили написания резюме сильно варьируются, всегда могут быть выделены общие блоки: Фамилия, Пол, Дата рождения, Описание предыдущих мест работы и т.д. Следовательно, мы локализуем общий набор атрибутов, которые могут быть представлены в большинстве резюме.

В целях придания модулю гибкости не используются неизменные шаблоны и правила извлечения данных из резюме и вакансий. Создаётся шаблон резюме и вакансии. Мы используем подход подгонки каждого резюме к созданному шаблону. Для некоторых конечных объектов должен быть создан набор правил для извлечения информации. Если некоторые блоки содержат некорректные отличия, модуль возвращается в обучающий режим и создаётся дополнительное правило извлечения информации, которое заносится в базу знаний. Когда встречается какой-либо новый блок резюме (например, информация о дополнительных интересах), в режиме правки шаблона добавляются новые элементы, а также создаётся правило извлечения. Далее все резюме обновляются со ссылкой на новое свойство. Такой модуль может быть применён для извлечения любых данных в сфере трудоустройства – резюме, анализа анкет и т.д.

3. Описание метода

3.1. Кластеризация резюме

Кластеризация – это процесс группировки данных в классы так, что объекты внутри кластера имеют высокую долю схожести по сравнению друг с другом, но они очень различаются с объектами других кластеров. Перечень классов определяется заранее и

включает все необходимые направления университетской подготовки студентов: менеджмент, мобильные коммуникации, компьютерные науки, радиоэлектроника и т.д. Каждое резюме после обработки представляется формой по такой схеме: ключ – значение $R = \{r_i\}$, где r_i – резюме, $r_i = \{<\text{ключ}, \text{значение}>\}$, где i=1 … n, n – количество атрибутов.

Описание модели одинаково для всех резюме. Для каждого кластера правило встречаемости резюме в определённой группе было определено как $F = \{f_{i1},...,f_{in}\}$.

Применяя данные условия, мы получим набор пересекающихся подмножеств $C_i \cap C_j \neq 0$, где $C_i = f_i(r_i)$.

Для каждого атрибута была применена мера TF-IDF [2,3]. Каждое резюме или вакансия d рассматривается как взвешенный вектор в пространстве элементов и каждый документ (вакансия или резюме) может быть представлен как $tf_1 * \log(n/df_1)$, $tf_2 * \log(n/df_2)$,..., $tf_m * \log(n/df_m)$, где tf_i – частота i-го элемента в документе и df_i – количество документов, содержащих i-й элемент, а n – общее число документов в выборке. Каждый вектор в документе должен быть нормализован, $\| d_{tfidfi} \|_2 = 1$.

Для оценки схожести применяется метод косинусов, который определяется как выражение:

$$\cos ine(d_i, d_j) = \left\langle d_i \bullet d_j \right\rangle \Big/ \left\| d_j \right\|_2 \times \left\| d_i \right\|_2 = \sum_{t=1}^{t} d_i \times d_j \Big/ \sqrt{\sum_{t=1}^{t} d_i^2} \times \sqrt{\sum_{t=1}^{t} d_j^2} \quad ,$$

где d_i и d_j – компоненты вектора документа; t – величина вектора.

Далее вычисляется суммарный TF-IDF вес для классификации резюме и вакансий.

3.2. Кластеризация резюме с помощью интегрированного подхода

Внутри каждого кластера мы определяем условия для разделения резюме и вакансий на подклассы, чтобы сгруппировать подкатегорию. Эти группирующие условия определяются пользователем. Другими словами, каждое резюме или вакансия – объект с атрибутами, где каждый атрибут – свойство резюме или вакансии, например, описание определённых навыков. На первом этапе кластеризации мы использовали иерархический подход [5,6]. Так создаётся иерархическая декомпозиция набора данных.

Мы объединяем иерархическую агломерацию и итеративно перераспределяем первым использованием иерархического агломеративного алгоритма с UPGMA методом [6,7], затем повышаем качество результата, используя итеративный подход [8,9], аналогично с кластеризацией методом Хамелеона [10]. На финальной итерации алгоритма определяется схожесть между каждой парой кластеров, учитывая как их относительную связанность между собой, так и их относительную схожесть.

В нашем алгоритме на протяжении первой фазы мы конструируем асимметричный k-NN граф с ребром между двумя точками, если для одной из них существует ближайший сосед среди всех соседей согласно величине k. Отметим, что вес ребра, соединяющего два объекта в k-NN графе, – это мера их схожести, как и простая мера расстояния (или обратная к их расстоянию).

Вес ребра мы вычисляем как взвешенное расстояние между объектами. В период фазы сжатия конструируется набор меньших гиперграфов. На первом этапе процесса сжатия мы выбираем множество вершин с максимальными степенями и сравниваем их с произвольным соседом. На других этапах мы рассматриваем каждую вершину в произвольном порядке и сравниваем их с соседней вершиной по ребру с наибольшим весом. Отметим, что обычно вес ребра, соединяющего два узла в сжатой версии графа – это количество рёбер в исходном графе, которые соединяют два множества исходных узлов, разделённых на два сжатых узла. В нашем случае мы вычисляем вес гиперребра как сумму весов всех рёбер, разбитых между собой в процессе шага сжатия. Мы останавливаем процесс сжатия на каждом уровне, как только количество мультивершин результирующего сжатого гиперграфа было уменьшено на константу менее двух. На следующем уровне алгоритма мы создаём множество малых гиперграфов, используя k-way многоуровневую парадигму [11].

Мы начинаем процесс декомпозиции с выбора k наиболее весомых мультивершин, где k = 8, 16, 32. После этого мы собираем одного за другим всех соседей для каждой из выбранных вершин и получаем начальную декомпозицию с ссылкой на уравновешивающую константу. Проблема вычисления оптимального деления пополам гиперграфа NP-сложна. Одна из наиболее часто используемых целевых функций – это минимизация декомпозиции пересечения гиперрёбер; т.е. суммарное число гиперрёбер, которые охватывают множественные разделения [11]. В наших экспериментах мы использовали «жадный» алгоритм повышения качества, разработанный Джорджем Каруписом [11], но как полезную функцию для каждой вершины мы вычисляли разницы между суммой весов рёбер, инцидентных вершине, которые ведут к другой секции, и суммой весов рёбер, которые остаются в секции. Мы выбирали вершину с максимальной полезной функцией и перемещали её, так что мы работали только с граничными вершинами.

После декомпозиции гиперграфа на множество малых частей мы начинаем соединять пару кластеров, для которых их относительные взаимосвязанность и близость высоки. В нашем исследовании для вычисления схожести между субкластерами была использована формула Джорджа Каруписа, модифицированное выражение с помощью изменения относительной взаимосвязанности на несколько выражений, которые проводят оценку средних весов рёбер в каждом подграфе и отношения количества рёбер, соединяющих две декомпозиции, к числу рёбер, находящихся внутри меньшей декомпозиции. Экспериментальные результаты показали, что этот метод не чувствителен к величине k и не нуждается в создании особого графа k-ближайших соседей [7].

Резюме считается рассмотренным, если подкластер определён и соответствующим образом сохранён в системе.

3.3. Аннотация резюме соискателя

В системах обработки текста используются различные подходы для текстовой аннотации. Наиболее распространённый способ – это список ключевых слов. Этот метод прост в реализации, но недостатком его является нехватка самоописательности. Другой путь – автоматическое создание аннотации. Он даёт достаточно качественный результат, но алгоритмически сложен. Рассматривая проблемную область, среди других методов, предлагаем аннотировать резюме кандидата с помощью добавления подмножества атрибутов из общих блоков всех резюме, таких как навыки и умения кандидатов.

Для наших экспериментов были вручную отобраны и помечены 200 резюме. Пометка включала расположение блоков «Образование», «Опыт работы» и «Другое». Поля «Контактная информация», «Хобби» и т.д. были внесены в блок «Другое». В этой части нашего эксперимента мы хотим создать список ключевых слов для каждого из описанных выше блоков. На начальном шаге обработки непечатные символы, стоп-слова, пометки и лишние пробелы, числа и аббревиатуры были удалены. Второй шаг – морфологический поиск. Применяется алгоритм «Porter stemming» в переводе на русский язык [12]. Он так же прост в применении, как и созданный на основе эвристических правил транзакций слов, и не требует поддержки словаря. К сожалению, при нетипичных словах он фиксирует ошибки, но это случается редко и не влияет на конечный результат. После процесса нормализации слово располагается в списке ключевых слов данного блока. Для сгенерированных ключевых слов вычисляется также их частота встречаемости. Иногда это ошибочно, и появляются строки длиной меньше трёх. Так как эти строки не относятся к семантическим носителям блока, можно их удалить. Как результат второго шага, мы получаем список основных ключевых слов с частотой их встречаемости в текстовом блоке. Одно слово может принадлежать нескольким спискам. В этом случае оно не является уникальной характеристикой блока. Для сохранения однозначности необходимо избавиться от пересечений ключевых слов. Сравниваются частоты встречаемости таких слов. Слово остаётся в той группе, где частота его встречаемости больше. Так мы получаем непересекающиеся множества ключевых слов с частотной характеристикой для каждой группы.

Как показывают практические эксперименты, использование только корней ключевых слов не даёт достаточно точного разделения. Поэтому чтобы определить границы блоков, были использованы фразы из их заголовков. На этапе ручного разбора поступающих файлов заголовки были выделены отдельно. Параллельно для каждого из блоков был

сформирован список ключевых слов и заголовков. Точность разделения блоков с помощью только ключевых слов составила 80%. Если граница заголовка не была известна, информация сохранялась в системе. По анализу заголовка мы определяли возможное разбиение текста резюме на блоки. Для одобрения или коррекции этого разбиения использовался статистический подход на основе доступной информации о ключевых словах. Как результат, мы получаем текст, разбитый на блоки «Образование», «Опыт работы» и «Другое». Разбиение текста на блоки происходит следующим образом. Для каждого слова мы находим его нормальную форму, используя уже рассмотренный алгоритм морфологического поиска [12]. Далее ищем полученную нормальную форму в списках ключевых слов, и если мы её находим – окрашиваем слово в цвет группы.

В компьютерных науках сущность – есть индивидуальная величина данных, созданных или используемых бизнес-процессом. Сущности нашей проблемной области – это дата рождения, супружеский статус, дата поступления и окончания одного либо нескольких высших учебных заведений, профессиональные навыки, гражданство, владение языками и т.д. Расположение сущностей производится по определённым правилам, созданным на основе стандартных выражений [13]. Они формируются на стадии обучения.

Использование резюме как общего шаблона позволяет достичь высокого качества классификации. Предложенный подход применён к анализу вакансий и позволяет решить такие проблемы, как сравнение резюме и существующих вакансий с помощью системы. Применяется метод составления списка топовых резюме (тех, по которым имеется наибольшее число запросов работодателей).

Разработана пробная версия системы. Анализ эффективности показал, что 500 резюме в 87% случаев были корректно разбиты по блокам, в 82% случаев факты были размещены корректно. Анализ случаев, когда система не смогла разбить текст по блокам правильно, показал, что это были нетипичные стили написания резюме, HTML-таблицы.

4. Заключение

Предложена идея разработки интеллектуальной системы поддержки процесса трудоустройства студентов. Являясь виртуальным консультантом по трудоустройству**,** такая система может существенно ускорить процесс поиска работы. Предложенный подход может быть использован для решения проблем классификации, сегментации и распределения фактов из других областей, связанных с документооборотом в сфере рекрутинга.

Универсальный шаблон резюме и вакансий позволяет достичь высокого качества классификации. Предложенный метод даёт возможность решать такие задачи, как аннотация резюме кандидатов и автоматическое сравнение резюме с существующими вакансиями. Предложен интегрированный кластерный подход оценки схожести резюме, на основании которого формируется перечень наиболее актуальных из них.

Список литературы: 1. *Liu, B., Lee, W. S., Yu, P., and Li, X.* 2002. Partially supervised classification of text documents. ICML-02. Salton, G. and McGill, M. (1983). Introduction to Modern Information Retrieval. McGraw-Hill. 2. *Yang, Y. and Pedersen J. P.* 1997. A comparative study on feature selection in text categorization. ICML-97. 3. *Andrew McCallum, Rosenfeld R., Mitchell T, Ng A.* 1998. Improving text clasification by shrinkage in a hierarchy of classes. In Proceedings of the International Conference on Machine Learning (ICML). P. 359-367. 4. *Toutanova K, Chen F, Popat K, and Hofmann Th.* 2001. Text classification in a hierarchical mixture model for small training sets. In Proceedings of the Tenth International ACM Conference on Information and Knowledge Management (CIKM). 5. *Joachims T.* 1997. A probabilistic analysis of the rocchio algorithm with TFIDF for text categorization , In Proc. Of the ICML'97. P.143-151. 6. *Zhao Y. and Karypis G.* 2002. Evaluation of hierarchical clustering algorithms for document datasets. In Proceedings of the International Conference on Information and ,Knowledge Management. 7. *Zhao Y. and Karypis G.* 2004. Empirical and theoretical comparisons of selected criterion functions for document clustering. Machine Learning, 55(3). 8. *Shatovska T., Safonova T., Tarasov I. A* Modified Multilevel Approach to the Dynamic Hierarchical Clustering for Complex types of Shapes. 2007. Lecture Notes in Informatics (LNI). Proceeding. Vol. P-107. P.176-186. 9. *Shatovska T., Safonova T., Tarasov I.* The New Software Package for Dynamic Hierarchical Clustering for Circles Types of Shapes. Proceedings of XIII-th International Conference KDS 2007, June, Varna, Bulgaria. P. 125-129. 10. *Karipys, G., Han, E.H., Kumar, V.* (1999). CHAMELEON: A hierarchical clustering algorithm using dynamic modeling, IEEE Computer: Special Issue on Data Analysis and Mining, 32(8), P. 68-75. 11. *Karypis G. and Kumar V.* 1999. Multilevel k-way hypergraph partitioning. In Proceedings of the Design and Automation Conference. 12. *Russian* stemming algorithm, 2005: http://

snowball.tartarus.org/algorithms/russian/stemmer.html. 13. *Keleberda I., Repka V., Biletskiy Y.* 2006. Building learner's ontologies to assist personalized search of learning objects. ICEC 2006. P. 569-573.

Поступила в редколлегию 16.05.2008

Шамша Борис Владимирович, канд.техн.наук, профессор кафедры ИУС ХНУРЭ. Научные интересы: разработка эффективных методов кластеризации в ИУС. Адрес: Украина, 61166, Харьков, пр. Ленина 14, тел. 702-14-51.

Шатовская Татьяна Борисовна, канд.техн.наук, доцент кафедры ПОЭВМ ХНУРЭ. Научные интересы: Data mining, Web mining, Text mining. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-14-46.

Гуд Анастасия Юрьевна, аспирантка кафедры ИУС ХНУРЭ. Научные интересы: разработка эффективных методов классификации в информационных управляющих системах. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-14-51.

УДК 519.23

В. М. ЛЕВЫКИН, Е. А. МОСПАН

РАЗРАБОТКА МОДЕЛИ ТЕХНОЛОГИИ ФОРМИРОВАНИЯ ЭЛЕКТРОННЫХ ДОКУМЕНТОВ В WEB-ОРИЕНТИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ

Описываются результаты исследования существующих технологий формирования электронных документов в WEB-ориентированных информационных системах, а также их обобщенная модель. В связи с выявленными недостатками подобных технологий разрабатывается модифицированная модель технологии формирования электронных документов с учетом специфики WEB-ориентированных информационных систем. Модифицированная технология предполагает формирование электронных документов на основании шаблона, представленного описательными языками, и разметки, а также выдачи выходного документа в формате, определенном пользователем.

1. Введение

В настоящее время электронные документы занимают важное место в области передачи информации и взаимодействия в человеко-машинных системах. Современные системы управления документами (Document Management System - DMS)[1], системы управления контентом (Content Management Systems - CMS)[2] и информационные системы (ИС) различных классов немыслимы без электронных документов. В первом случае электронные документы составляют основу организации документооборота, во втором являются одним из основных продуктов функционирования системы, который может выступать не только вариантом представления данных системы, но и юридическим документом. В связи с этим задача формирования и поддержки выходных документов возникает практически у каждого разработчика ИС.

2. Актуальность исследования

Формирование электронного документа в ИС связано с конкретной функциональной задачей или бизнес-функцией, а именно с входными и выходными данными, относящимися к ним. Процесс внесения изменений данных, необходимых для документа, неразрывно связан с изменениями требований к самой задаче. Однако подобные изменения не означают внесения корректировок в структуру электронного документа, которая в свою очередь может изменяться вне зависимости от реализации требуемой задачи. Добавление новых требований к структуре чаще всего определяется заказчиком. Исходя из этого, выходной электронный документ необходимо рассматривать как совокупность его структуры и данных, которые определяются состоянием системы. Под структурой документа будем понимать его шаблон, который определяет форматирование и представление выходного документа. Под данными будем понимать набор входных и выходных данных функциональной задачи, которые необходимо отразить в создаваемом документе.