Система поддержки принятия кадровых решений на основе векторной модели ИТ-специалиста

Р. А. Файзрахманов[1], Д. В. Яруллин[2], П. Ю. Фоминых[3]
Пермский национальный исследовательский политехнический университет
[1]fayzrakhmanov@gmail.com, [2]d.v.yarullin@ya.ru, [3]phominykh1997@gmail.com

***Аннотация*. Работа описывает подход к выбору индивидуального направления профессиональной подготовки или переподготовки в ИТ-индустрии для соискателя, уже обладающего какими-либо компетенциями в указанной предметной области. Описывается система поддержки принятия решений, позволяющая получить список востребованных в настоящее время ИТ-специализаций в соответствии с текущими компетенциями пользователя. Оценка востребованности специализаций в течение определенного периода производится на основе анализа компетенций в текстах ИТ-вакансий, опубликованных на сайтах-агрегаторах и сгруппированных по регионам. С помощью кластеризации определяются наборы взаимосвязанных компетенций, интерпретируемые как рабочие нагрузки для различных ИТ-специализаций, востребованных в регионе в указанный период. Модель реализуется в прототипе системы поддержки принятия решений. Пользователь системы указывает свои текущие компетенции и интересующие его регион и период; на основе введенных данных система подбирает наиболее релевантные для пользователя рабочие нагрузки. Вывод системы может быть использован для поддержки принятия кадровых решений относительно наиболее подходящей роли или специализации для кандидата, обладающего определенным набором компетенций.**

Ключевые слова: система поддержки принятия решений, множество рабочих нагрузок, кластеризация, интеллектуальный анализ данных, извлечение сущностей

I. Постановка проблемы

В настоящее время одной из основных проблем на рынке труда является выбор подходящей для соискателя специализации в рамках его предметной области или профессии. Соискатель, обладающий определенными компетенциями, может испытывать затруднения при выборе конкретной специализации, сталкиваясь с тем, что не все требуемые навыки и компетенции явно перечислены в текстах вакансий. Также вакансии на одну и ту же должность, например, «Веб-программист», могут требовать от соискателя различных компетенций. Для решения этой проблемы в первую очередь необходимо понять, какие компетенции необходимы для той или иной специализации, как именно рынок труда представляет такого специалиста.

Работа ставит своей целью создание системы поддержки принятия решений по индивидуализированной профессиональной подготовке или переподготовке на основе модели ИТ-специалиста. Авторы фокусируют свое внимание на программистах как одной из наиболее «гибких» профессий в данной области с большим числом различных специализаций.

В рамках исследования также предлагается метод сбора данных вакансий. Компетенции извлекаются из текстов вакансий с использованием методов обработки естественного языка и векторизуются для последующего кластерного анализа. Кластерный анализ позволяет определить, структурировать и упорядочить наборы взаимосвязанных компетенций, интерпретируемые в дальнейшем как рабочие нагрузки для различных ИТ-специализаций, востребованных в регионе в течение определенного периода. Вслед за существующими исследованиями в данной области, минимальную единицу нашей модели мы обозначаем как «компетенция» или «навык» [1].

В данной работе описывается прототип системы поддержки принятия решений с графическим интерфейсом, предлагающей наиболее релевантные для пользователя рабочие нагрузки, тестирование приложения и интерпретация полученных результатов.

II. Сбор и подготовка данных

В качестве основного источника данных предлагается рассматривать требования работодателей к компетенциям специалистов на определенных должностях. Требования могут быть извлечены из текстов вакансий, публикуемых на сайтах-агрегаторах. Преимуществами данного источника данных является возможность автоматизации сбора данных и мониторинга изменений в динамике с целью корректировки индивидуального плана подготовки в соответствии с текущими спросом на рынке труда.

В данной работе рассматриваются требования работодателей в двух странах: России и Германии. Для России выбран агрегатор вакансий «HH.ru», а для Германии — «Monster.de». Оба сайта предоставляют доступ к тексту вакансий через API [2]. Поиск ограничен по регионам России и федеральным землям Германии с возможностью анализа данных по каждому региону в отдельности.

Для анализа текстов вакансий были применены методы обработки естественного языка (NLP) [3], [4]. Для нормализации русских слов используется модуль pymorthy2 с пакетами словарей OpenCorpora [5]. На сайте «Monster.de» публикуются вакансии на английском языке, поэтому для нормализации и препроцессинга английских

лексем используется Natural Language Toolkit (NLTK). После нормализации из текстов удаляются стоп-слова [6].

Затем происходит извлечение навыков. Список навыков сформирован на основе метаданных с сайта HH.ru. Навыки также нормализуются. Для поиска навыков в предварительно обработанных текстах вакансий используется метод *n*-грамм [7]. Метод *n*-грамм позволяет учитывать не только слова, но и фразы, что позволяет извлекать навыки, состоящие из более чем одного слова (например, «1С программирование», «Entity framework»). Эмпирически установлено, что максимальное число лексем в одном навыке в данной предметной области равно шести.

После извлечения навыков для каждого региона необходимо преобразовать их естественно-языковую форму в векторную репрезентацию для дальнейшей обработки.

III. Построение модели

Для выявления взаимосвязанных компетенций предложен кластерный анализ как средство группировки навыков, наиболее часто встречающихся в одном контексте. Каждая группа рассматриваются как репрезентация рабочей нагрузки и стека технологий (понимаемый как набор инструментов, применяющийся при работе в проектах и включающий в себя определенные языки программирования, фреймворки, СУБД и т. д.) для определенных ИТ-специализаций в регионе (например, «веб-программирование», «1С программирование», «разработка мобильных приложений» и т. п.).

В рамках данного исследования было рассмотрены несколько подходов к выделению и группировке взаимосвязанных компетенций.

Изначально были рассмотрены методы векторизации и группировки непосредственно текстов вакансий. Была выдвинута гипотеза о том, что навыки могут также являться признаками, по которым вакансии на схожие должности могут быть сгруппированы. Были проведены эксперименты по кластеризации с использованием word2vec [8] и тематического моделирования алгоритмом неотрицательного матричного разложения [9].

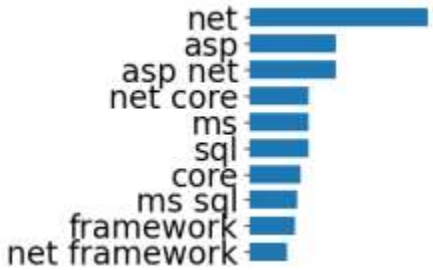

Рис. 1. Тема, определенная как «.NET», регион Санкт-Петербург

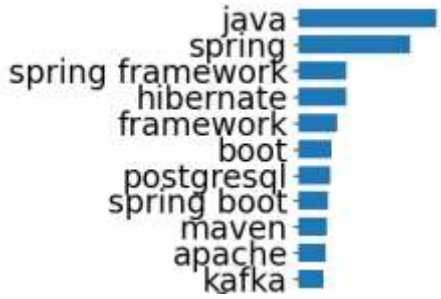

Рис. 2. Тема, определенная как «Java», регион Санкт-Петербург

Рис. 3. Тема, определенная как «Веб-разработка», регион Санкт-Петербург

Хотя ключевые слова для ряда тем (представлены на рис. 1–3) были выделены корректно и действительно являлись искомыми навыками, многие темы характеризовались нерелевантными частотными словами, действительно являющимися ключевыми для данной группы текстов, но при этом никак не связанных с решением нашей задачи по выделению взаимосвязанных навыков (пример такой темы приведен на рис. 4).

Рис. 4. Тема, определенная как «Мотивационные сообщения», регион Санкт-Петербург

Аналогичным недостатком в рамках решения нашей задачи обладает word2vec, в основе которого лежит поиск окружающих слов. Данный метод подходит для обработки текстов, но не пригоден для выделения специфических сущностей (навыков) из одной группы. В описании вакансий в окружении слов-навыков частотными являются общие слова, такие как «команда», «уверенный», «владение».

Затем был рассмотрен обратный подход, когда векторизуются не тексты, а сами навыки. Признаками в этом случае являются сами вакансии. Метод преобразования в бинарный вектор заключался в нахождении гиперплоскости в пространстве признаков, разделяющей набор на две части: одна содержит все

положительные примеры (навык присутствует в текстах данного региона), а другая — все отрицательные (навык отсутствует в текстах региона) [10].

Полученные векторы навыков были кластеризованы при помощи ряда алгоритмов, включая *k*-средние, иерархическую кластеризацию Уорда, спектральную кластеризацию [11]. Все перечисленные подходы требуют заранее определенное число кластеров, что оказалось затруднительным на реальных данных. В крупных регионах (например, Москва) может быть около 2000 текстов за один период, тогда как в более маленьких (Иркутская область) их может быть только 113.

В качестве альтернативного подхода был предложен алгоритм распространения близости, поскольку он автоматически определяет количество кластеров в зависимости от набора данных. Алгоритм предполагает обмен сообщениями между всеми точками данных. Расчет продолжается чередованием двух шагов передачи сообщений, которые обновляют матрицы «ответственности» и «доступности».

Итерации выполняются до тех пор, пока либо не останутся неизменными границы кластера в течение нескольких итераций, либо не будет достигнуто максимально допустимое количество взаимодействий [12].

В качестве меры близости выбрано косинусное сходство. Косинусное сходство отражает коэффициент корреляции между двумя векторами. Эта мера часто используется для текстовых данных.

IV. Система поддержки принятия решений

На основе описанного подхода был разработан прототип системы поддержки принятия решений. Система включает в себя вывод всех возможных рабочих нагрузок для региона в указанный период (рис. 5, 6) и модуль подбора рабочих нагрузок, которые подходят пользователю (рис. 7).

Вывод множества рабочих нагрузок доступен для двух стран: Россия и Германия. Выбор страны определяет доступные регионы. Множества рабочих нагрузок выводятся по месяцам. Доступен массив данных с марта 2020 года, по умолчанию выбран на последний доступный месяц. После выбора необходимых параметров формируются результаты кластеризации. Результаты кластеризации выводятся с информацией о количестве обработанных вакансий, количестве выделенных навыков и кластеров.

Вывод имеет специальный формат: синий цвет для центров кластеров, фиолетовый цвет для компетенций, которые пользователь уже имеет, черный цвет для навыков, которыми пользователю необходимо овладеть для этой конкретной рабочей нагрузки.

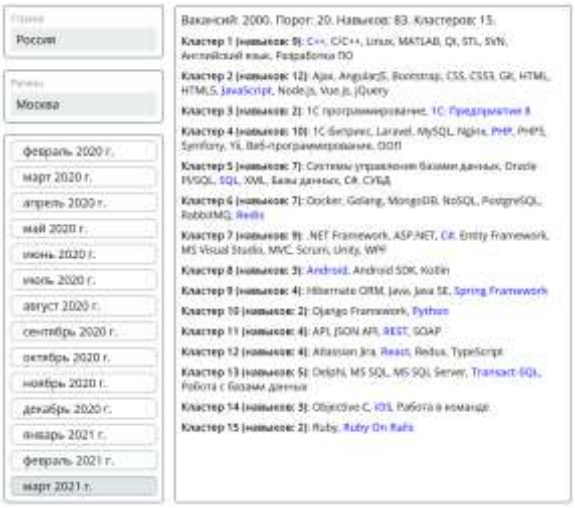

Рис. 5. Список рабочих нагрузок для Москвы на начало марта 2021 г.

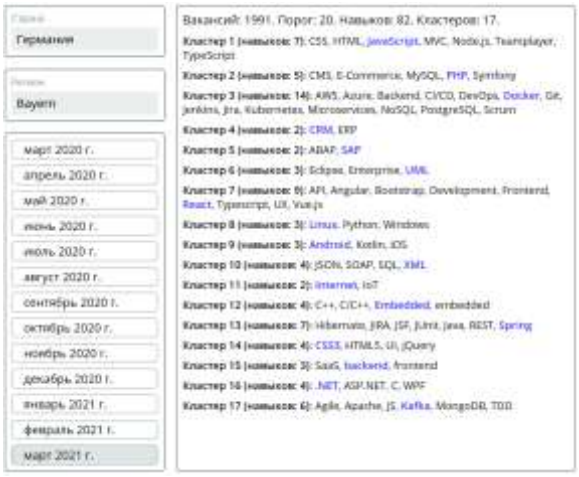

Рис. 6. Список рабочих нагрузок для Баварии на начало марта 2021 г.

Рис. 7. Модуль подбора индидвиуальных рабочих нагрузок

Для использования модуля пользователь должен ввести навыки, которыми он владеет, разделить их запятыми, выбрать страну, регион и дату сбора вакансий. На основе заданной фильтрации при нажатии кнопки «Подобрать» появляются рабочие нагрузки для региона, содержащие введенные навыки.

Рис. 8. Пример для Санкт-Петербурга

Первый пример (рис. 8) содержит следующие настройки: Россия, Санкт-Петербург, март 2021 года. Были введены навыки «Java», «Git», в ответ были получены два кластера. Множество рабочих нагрузок, формируемое в результате кластеризации, определяет конкретные релевантные специализации по совокупности необходимых навыков. На основании данных о компетенциях можно сформировать рекомендации для плана дальнейшей подготовки или переподготовки.

Рассмотрим, как специализация определяется ее множеством рабочих нагрузок. Для этого примера были введены следующие настройки: Россия, Свердловская область, ноябрь 2020 года (рис. 9). Введены компетенции «Angular», «Kotlin», «iOS». В результате были сформированы 2 кластера, подходящие для этих параметров, поскольку навыки «Kotlin» и «iOS» включены в один кластер. Рассмотрим этот кластер. Он включает следующие компетенции: «Android», «Android SDK», «Java», «Kotlin», «iOS». Центром кластера является «Android». В совокупности все эти навыки определяют профессию мобильного разработчика с фокусом на ОС Android.

Рис. 9. Пример для Свердловской области

Таким образом, на основе множества рабочих нагрузок была определена профессия разработчика Android, что подразумевает знание операционной системы Android, знание языков программирования Kotlin и Java, а также умение работать в среде разработки Android SDK.

По данному запросу пользователю может быть рекомендовано освоить профессию разработчика Android, так как большинство навыков, которыми он обладает, необходимы для этой профессии. В качестве индивидуального учебного плана система рекомендует изучение языка программирования Java и освоение среды разработки Android SDK.

V. Результаты

В результате исследования был предложен подход, позволяющий выявить наиболее востребованные навыки ИТ-специалистов и тенденции их развития.

Предложен алгоритм извлечения компетенций, необходимых работодателю, из текстов вакансий. С помощью алгоритмов обработки естественного языка и методов кластерного анализа генерируются множества рабочих нагрузок для различных регионов.

Результатом работы стал прототип системы поддержки принятия решений, позволяющий пользователю определить конкретную специализацию в ИТ-сфере с учетом имеющихся у него навыков. На основе информации о перечне компетенций, которыми должен обладать специалист в выбранной области, может быть составлен индивидуальный план подготовки или переподготовки.

Список литературы

[1] Nikulchev E., Ilin D., Matishuk E. Scalable Service for Professional Skills Analysis Based on the Demand of the Labor Market and Patent Search. Procedia Computer Science. Volume 103, 2017. Pages 44-51. DOI: https://doi.org/10.1016/j.procs.2017.01.008

[2] HeadHunter API: документация и библиотеки // Электронный ресурс. URL: https://github.com/hhru/api, дата обращения: 12.03.2021

[3] Bird S., Loper E., Klein E. Natural Language Processing with Python. O'Reilly Media Inc. 2009.

[4] Goldberg Y. Neural Network Methods for Natural Language Processing, vol. 10, no. 1 , 2017, pp. 1-309.

[5] Korobov M.: Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts, 2015. Pp 320-332. DOI: https://doi.org/10.1007/978-3-319-26123-2_31

[6] Kim S., Gil J. Research paper classification systems based on TF-IDF and LDA schemes, vol. 9, no. 30, 2019, pp. 9-30.

[7] Jurafsky, D., Martin, J. H. Speech and Language Processing, 2019, pp. 1-621.

[8] Mikolov T., Sutskever I., Chen K., Corrado G.S., Dean J. Distributed representations of words and phrases and their compositionality // Advances in neural information processing systems, vol. 26, 2013, pp.3111-3119.

[9] Wang Y.-X., Zhang Y.-J., Nonnegative matrix factorization: A comprehensive review // IEEE Transactions on Knowledgeand Data Engineering, vol. 25, no. 6, 2012, pp. 1336–1353.

[10] Burges C. A tutorial on support vector machines for pattern recognition. Volume 2, 1998. Pages 955-974

[11] Wierzchon, S., Klopotek, M., Modern Algorithms of Cluster Analysis, 2018, pp. 1-975.

[12] Thavikulwat P. Affinity propagation: a clustering algorithm for computer-assisted business simulations and experiential exercises. Volume 35, 2008, Pages. 220-224.