

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК
Кафедра программной и системной инженерии

РЕКОМНДОВАНО К ЗАЩИТЕ В ГЭК
И ПРОВЕРЕНО НА ОБЪЕМ
ЗАИМСТВОВАНИЯ

Заведующий кафедрой

д.т.н., профессор


А.Г. Ивашко

2018 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОГО АССИСТЕНТА
СПЕЦИАЛИСТА ОТДЕЛА ПРОДАЖ

09.04.03. Прикладная информатика

Магистерская программа «Прикладная информатика в экономике»

Выполнил работу
Студент 2 курса
Очной формы обучения



Миллер
Юрий
Олегович

Научный руководитель
Доцент кафедры ИС, к.ф.н.



Бидуля
Юлия
Владимировна

Рецензент
Заместитель генерального директора
ООО «Ойл Линк»



Епифанов
Виктор
Сергеевич

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК
Кафедра программной и системной инженерии

РЕКОМНДОВАНО К ЗАЩИТЕ В ГЭК
И ПРОВЕРЕНО НА ОБЪЕМ
ЗАИМСТВОВАНИЯ
Заведующий кафедрой
д.т.н., профессор
_____ А.Г. Ивашко
_____ 2018 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОГО АССИСТЕНТА
СПЕЦИАЛИСТА ОТДЕЛА ПРОДАЖ

09.04.03. Прикладная информатика

Магистерская программа «Прикладная информатика в экономике»

Выполнил работу
Студент 2 курса
Очной формы обучения

Миллер
Юрий
Олегович

Научный руководитель
Доцент кафедры ИС, к.ф.н.

Бидуля
Юлия
Владимировна

Рецензент
Заместитель генерального директора
ООО «Ойл Линк»

Епифанов
Виктор
Сергеевич

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
ЦЕЛИ И ЗАДАЧИ.....	4
ГЛАВА 1. ОПИСАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....	5
ГЛАВА 2. ОБЕСПЕЧЕНИЕ НАДЕЖНОСТИ И БЕЗОПАСНОСТИ СИСТЕМЫ.....	17
ГЛАВА 3. ПРОВЕДЕНИЕ ИСПЫТАНИЙ СИСТЕМЫ	19
ГЛАВА 4. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ.....	24
ГЛАВА 5. ОЦЕНКА ОРГАНИЗАЦИОННОЙ ЭФФЕКТИВНОСТИ ПП	27
ЗАКЛЮЧЕНИЕ	28
СПИСОК ЛИТЕРАТУРЫ	29
ПРИЛОЖЕНИЕ 1	341
ПРИЛОЖЕНИЕ 2.....	33
ПРИЛОЖЕНИЕ 3.....	57
ПРИЛОЖЕНИЕ 4.....	62
ПРИЛОЖЕНИЕ 5.....	63
ПРИЛОЖЕНИЕ 6.....	64

ВВЕДЕНИЕ

На сегодняшний день темпы развития бизнеса и торговых отношений диктуют новые формы ведения работы, при которых интеллектуальные ассистенты специалистов становятся неотъемлемым атрибутом операционной деятельности любой организации.

Интеллектуальные ассистенты представляют собой интегрированные в среду работы специалистов интеллектуальные системы (ИС, англ. intelligent system) — это технические или программные системы, способные справляться с задачами, традиционно являющимися творческими, а решение подобных задач считается сугубо человеческой прерогативой. К подобным задачам относятся, например, такие, как задачи распознавания и классификации.

ООО «Ойл Линк» занимается подбором и поставкой оборудования и комплектующих для буровых установок различного производства. Рассматривая работу с коммерческими предложениями для клиентов (далее КП), можно выявить, что используемое при подборе номенклатуры программное решение подразумевает сугубо ручной подбор экспертом-специалистом. Такой способ основывается только на знаниях специалиста и каталоге существующей номенклатуры, что не учитывает предыдущий опыт подбора для конкретных клиентов, а также не результативно в случае полного несоответствия формулировок содержания известным номенклатурам.

Подбор номенклатуры нуждается в совершенствовании путем разработки и внедрения интеллектуального ассистента специалиста, называемого далее «И.Т.А.». Система, автоматически подбирающая номенклатуру исходя содержания заказа клиента с учетом его предыдущих заказов позволит высвободить до нескольких десятков человеко-часов в месяц среди специалистов отдела продаж.

ЦЕЛИ И ЗАДАЧИ

Цель работы: высвобождение до 40 человеко-часов в месяц (среди специалистов отдела продаж) за счёт автоматизации функции подбора номенклатуры.

Достижение цели предполагает разработку интеллектуального ассистента специалиста отдела продаж (системы автоматического подбора) и апробацию системы на базе организации-заказчика.

Задачи:

- Разработка программного обеспечения системы
- Разработка механизма ранжирования для автоматического подбора
- Обеспечение надежности и безопасности системы
- Проведение испытаний системы
- Написание руководства пользователя
- Оценка организационно-экономической эффективности ПП

ГЛАВА 1. ОПИСАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

1.1 Функционал системы

Функционал системы определен необходимостью сокращения трудозатрат специалиста отдела продаж при подборе номенклатуры содержанию заказа в документе коммерческого предложения (далее КП; в среде пользовательской конфигурации 1С: Управление Торговлей) посредством внедрения автоматического интеллектуального подбора. Внешний вид интерфейса пользовательской конфигурации до внедрения системы проиллюстрирован на рисунке 1. Специалист вводит ручную значение колонки «Содержание» в качестве заказа клиента, затем самостоятельно заполняет поле «Номенклатура», подбирая объект из справочника, опираясь на экспертный опыт и учитывая предыдущие заказы данного клиента.

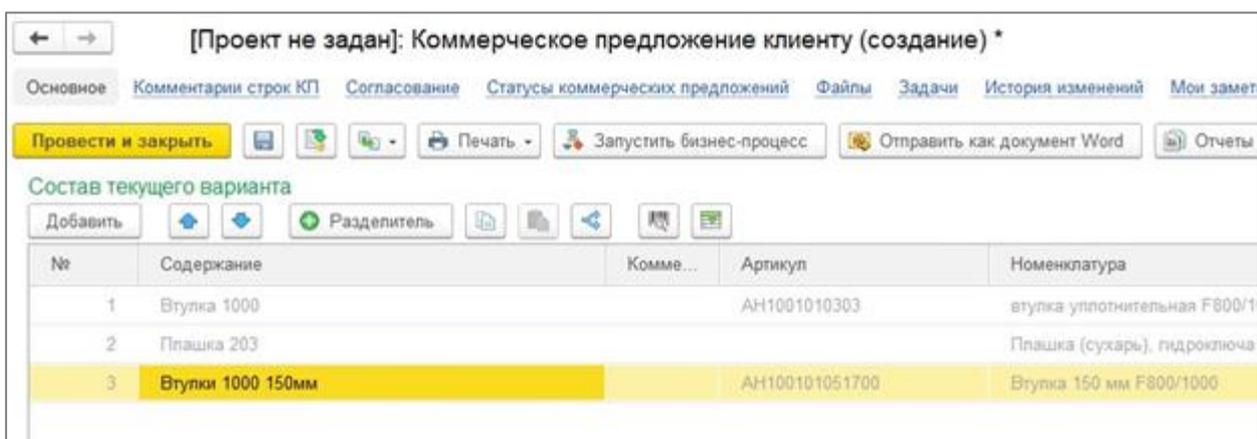


Рисунок 1. Интерфейс редактирования документа КП

Общей функцией системы является подбор номенклатуры для содержания заказа документа КП в интерфейсе пользовательской конфигурации 1С. Содержание заказа имеет вид строки на естественном языке. Подбор основан на анализе ранее заполненных КП (т.е. существуют товарные позиции с заполненными полями «Содержание» и «Номенклатура»), а также сопоставлении слов, входящих в содержания и названия номенклатур. Функционал сочетает в себе две функции системы, доступных специалисту

отдела продаж в среде пользовательской конфигурации 1С (при редактировании документа КП):

- Стандартный подбор (кнопка «Подобрать через И.Т.А.»)
Описание: подбор и заполнение одной наиболее подходящей с точки зрения механизма ранжирования номенклатуры для каждого содержания

- Расширенный подбор (кнопка «Расширенный подбор»)
Описание: подбор нескольких наиболее подходящих с точки зрения механизма ранжирования номенклатур для одного конкретного (выделенного) содержания с выводом их в виде списка для выбора

При этом нужно понимать, что в нормальном режиме (с 07:00 по 00:00 ежедневно) система функционирует следующим образом:

1. Пользователь (специалист отдела продаж) при добавлении или редактировании документа КП заполняет содержания заказа в табличной части вкладки «Товары» (товарные позиции на естественном языке) и по желанию указывает клиента в поле «Клиент». Затем, пользователь нажимает на кнопку «Подобрать через И.Т.А.» или на кнопку «Расширенный подбор через И.Т.А.», чтобы автоматически заполнить поле «Номенклатура», либо соответственно, чтобы получить список наиболее подходящих кандидатов-номенклатур в отдельном окне (Подробнее в руководстве пользователя в п. 5).

2. Конфигурация пользователя получает значение ID (идентификатора) клиента на основании заполненного пользователем поля «Клиент» и значения позиций колонки «Содержание» из табличной части вкладки «Товары» в следующем виде. Полученные строки в единую строку с разделителями.

3. Конфигурация пользователя передает внешнему программному модулю данные из пункта 2 (посредством GET http-запроса к веб-серверу, поддерживаемым внешним ПМ; подробнее взаимодействие отражено в главе 1.5)

4. Система на стороне внешнего ПМ принимает строку и запускает в работу механизм ранжирования для подбора номенклатуры (подробнее механизм ранжирования отражен в главе 1.6) и возвращает полученный результат подбора в виде строки

5. Система на стороне пользовательской конфигурации 1С разбивает полученную в пункте 4 строку на ID (идентификаторы) подобранной номенклатуры и заполняет значения колонки «Номенклатура» табличной части вкладки «Товары», подставляя номенклатуру по идентификатору из справочника «Номенклатура». Подстановка производится по порядку, сохраняя соответствие содержания номенклатуре (рисунок 2). При изначальном выборе расширенного подбора полученные идентификаторы номенклатуры будут выведены в виде элементов списка «Номенклатура» в отдельном окне «Расширенный подбор через И.Т.А» для выбора (внешний вид такого окна представлен на рисунке в главе 4).

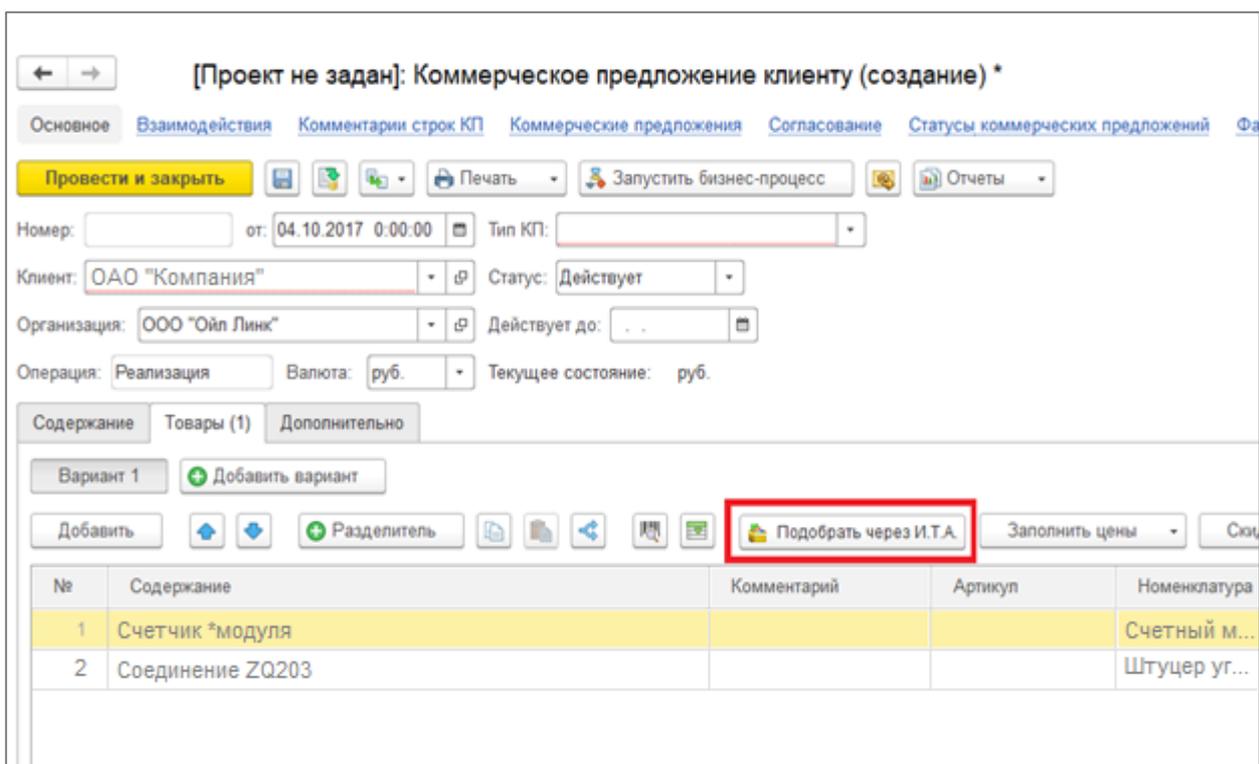


Рисунок 2. Интерфейс пользовательской конфигурации 1С в результате функционирования стандартного подбора

В режиме обучения (с 00:00 по 07:00 ежедневно) система функционирует следующим образом:

- Сервер 1С выгружает на файловое хранилище файлы содержаний и номенклатуры (структуры данных в приложении 1) в 00:00
- В 00:30 Производится перезагрузка ОС внешнего ПМ согласно планировщику задач
- По выполнению перезагрузки производится запуск системы согласно планировщику задач
- Внешний ПМ копирует файлы содержаний и номенклатуры из файлового хранилища

Описание операций и листинги программного кода, задействованные в обучающем режиме, содержатся в п. 3.

1.2 Состав модулей системы

- 1. Модуль 1С (Доработанная конфигурация 1С)
 - 1.1 Доработанная пользовательская конфигурация клиента 1С (Доработанный интерфейс документа КП (при добавлении КП и редактировании КП))
 - 1.2 Доработанная конфигурация сервера 1С (Механизм выгрузки – регламентное задание)
- 2. Внешний ПМ
 - 2.1 Механизм загрузки данных из файлового хранилища организации
 - 2.2 Веб-сервис для ИО с пользовательской конфигурацией 1С
 - 2.3 Алгоритм автоматического подбора (механизм ранжирования)

Состав модулей, включая сервер 1С, пользователей, файловое хранилище организации (включая выгрузку данных в виде файла формата .csv) и внешний ПМ, проиллюстрирован на рисунке 3.

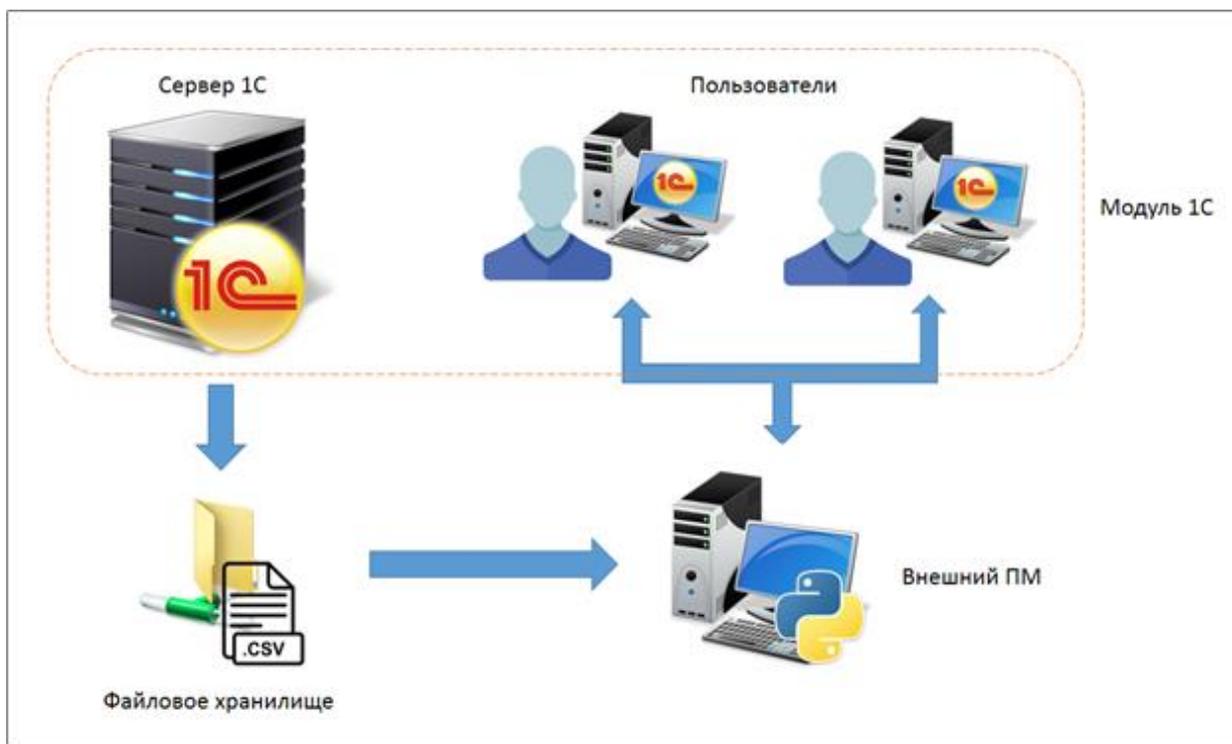


Рисунок 3. Схема модулей системы

1.3 Выбор средства разработки

В качестве средства разработки системы (помимо доработки конфигурации 1С Управление Торговлей) был выбран язык Python и среда разработки PyCharm Community Edition.

Основные преимущества языка Python заключаются в высокой скорости разработки (объем программного кода на языке Python обычно составляет около третьей части эквивалентного программного кода на языке Java или C#, что означает меньший объем ввода с клавиатуры, меньшее количество времени на отладку и меньший объем трудозатрат на сопровождение системы). Кроме того, в составе Python поставляется множество бесплатных модулей (библиотек). Обработка естественного языка является классической задачей, решаемой с помощью средств языка Python, что подразумевает собой не только специализированные модули, но и развитое многоязычное сообщество.

Состав модулей, примененных в разработке системы:

- Bottle – модуль для создания веб-сервера, способного принимать http-запросы и отправлять ответ в определенном виде с возможностью настройки маршрутизации
- Re – модуль для работы с регулярными выражениями. Модуль используется в системе для нахождения в строке содержания характеристики размера номенклатуры вида $A \times B$, $A \times B \times C$ и т.д., где A, B, C – значения измерений целого и вещественного типа
- Shutil – модуль, содержащий функции для обработки файлов и папок. Модуль используется в системе для копирования файлов выгрузки «Содержания.csv» и «Номенклатура.csv» из файлового хранилища в папку внешнего ПМ (проекта PyCharm)
- Codecs – модуль для чтения и записи файлов с настройками кодировки. Модуль используется в системе для открытия файлов выгрузки «Содержания.csv» и «Номенклатура.csv» в кодировке UTF-8.

- Stemmer – модуль стемминга, поддерживающий русский язык. Модуль используется в системе для преобразования слов к канонической форме (отсечение окончаний и суффиксов) для успешного поиска совпадения между строками в разрезе слов.

1.4 Доработка конфигурации 1С

В ходе работы были произведены доработки серверной и пользовательской конфигураций 1С:Предприятие (УТ), развернутой в стенах организации ООО «Ойл Линк».

Доработка конфигурации сервера 1С:

- Новый объект типа «Регламентное задание» - задание на выгрузку данных ранее проведенных в системе документов КП и данных справочника номенклатуры. Структура выгружаемых данных в формате «.csv» с разделителем элементов «табуляция» отражена в приложении 1.1 и 1.2. Расписание задания: по времени, каждый день в 00:00.

Доработка формы документа «Коммерческое предложение»:

- Новые кнопки «Подобрать через И.Т.А.» и «Расширенный подбор с И.Т.А.» во вкладке «Товар», включая модуль обработки нажатия каждой кнопки (рисунок 4).

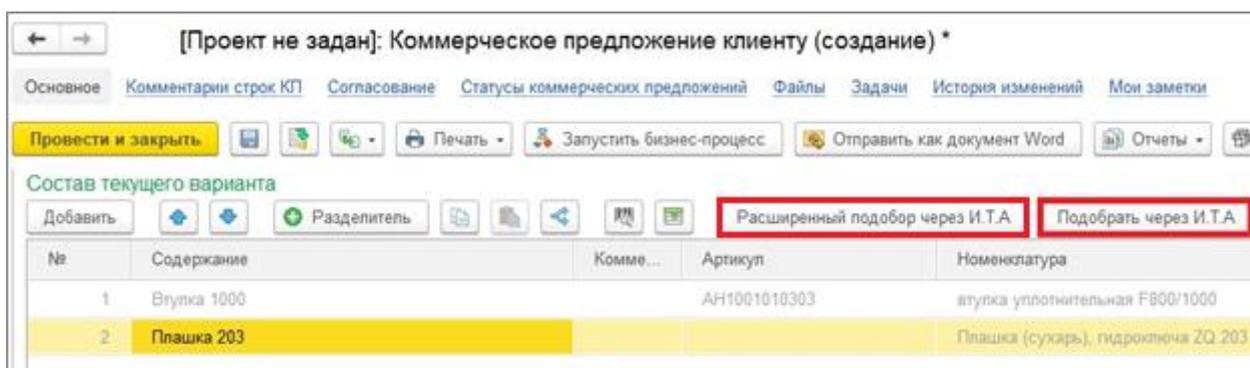


Рисунок 4. Новые элементы в интерфейсе пользовательской конфигурации

Отправка http-запросов к веб-серверу внешнего модуля по нажатию кнопок реализована с использованием следующих объектов 1С:

- *HTTPСоединение*
- *HTTPЗапрос*

Пользовательская конфигурация 1С отправляет запрос на адрес, приведенный далее в главе 1.5 и получает ответ с помощью вызова функции:

➤ *HTTPСоединение.ОтправитьДляОбработки()*

• Новая форма «Расширенный подбор через ИТА» для выбора элемента из таблицы результатов работы расширенного подбора (рисунок 5), включающее таблицу значений с полями «Артикул», «Номенклатура» и кнопка «Выбрать», передающая номенклатуру в выделенную строку табличной части формы документа.

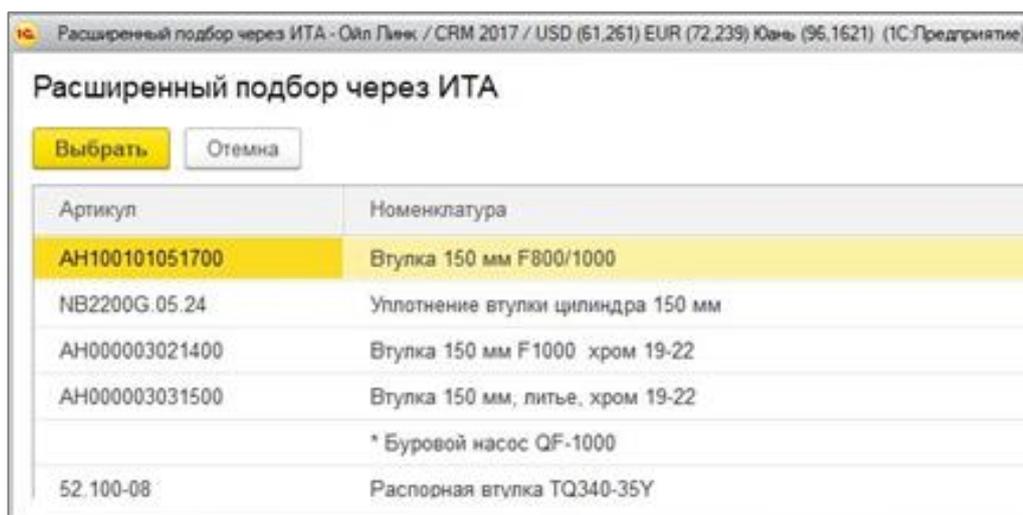


Рисунок 5. Окно для выбора элемента из списка результатов подбора

Данная новая форма является пользовательской и была создана с помощью встроенного конструктора форм 1С. Открытие формы выполняется с использованием функции:

➤ *ОткрытьФорму()*

1.5 Реализация информационного обмена

Информационный обмен между пользовательской конфигурацией 1С и внешним программным модулем на языке Python заключается в передаче данных о клиенте и содержании его заказа и возврате подобранной номенклатуры в качестве результата подбора.

Обмен данными производится согласно структуре данных в приложении 1.3 и 1.4 и реализован с использованием http веб-сервера и модуля Bottle. Модуль bottle представляет собой бесплатный инструмент для создания и

поддержании работы веб-сервера на языке Python, способного принимать http-запросы и отправлять ответ в определенном виде с возможностью настройки маршрутизации.

С помощью модуля bottle настроена маршрутизация следующего вида:

- 10.1.124.30:8082/std/<order:path>
- 10.1.124.30:8082/ext/<order:path>

Где: 10.1.124.30:8082 – адрес ПК с веб-сервисом внешнего ПМ в локальной сети организации, std – маршрут для стандартного подбора, ext – маршрут для расширенного подбора, <order:path> - строка, передаваемая конфигурацией 1С (содержит данные согласно структуре в приложении 1.3, обрабатывается сервером как строка с игнорированием символа ‘/’). Программный код с настройками веб-сервера приведен в листинге 1.

Листинг 1. Программный код веб-сервера

```
from bottle import route, run, template
...
@route('/std/<order:path>')
def index(order):
    return work(order)
@route('/ext/<order:path>')
def index(order):
    return work_ext(order)
...
run(host='0.0.0.0', port=8082)
```

Пример обращения к веб-серверу:

- «http://10.1.124.30:8082/ext/УТ-0000092|Счетчик *модуля»

Пример ответа веб-сервера в качестве результата механизма ранжирования (ID подобранной номенклатуры для 1С через разделитель '|'):

- «УТ-00007079|УТ-00007080»

Способ информационного обмена между сервером 1С и внешним ПМ подробнее отражен в главе 3, листинг 4.

1.5 Разработка механизма ранжирования

Механизм ранжирования широко применяем в информационном поиске, в т.ч. поисковыми системами [1]. В основе ранжирования любых документ лежит релевантность - степень найденного документа исходному запросу пользователя. Чем выше релевантность, тем выше документ будет находиться

в поисковой выдаче, а значит, более соответствует пользовательскому запросу [2]. Степень данного соответствия напрямую определяется алгоритмом расчёта релевантности и заранее определенными факторами. Так, ранжирование в поисковой выдаче Яндекса применительно к веб-сайтам определяют такие факторы, как возраст домена, формат страницы, ключевое вхождение в имени домена, наличие всплывающих окон и экспертная оценка ассессоров (люди, оценивающие параметры веб-сайта и качество самой поисковой выдачи).

В ходе работы была произведена предварительная реализация подбора с использованием векторной модели. Основопологающим этапом такой реализации является векторизация содержания, к которому требуется подобрать номенклатуру. Каждая строка содержания приводится к векторному виду на основании наличия или отсутствия в строке каждого из 300 элементов словаря. Так, например, если в словаре первые слова: «счетчик», «соединен», «кольцо», «модуль», «zq203» ..., строки содержаний примут следующее представление

➤ 1 0 0 1 0 ... (ряд чисел)

Предварительное тестирование работы алгоритма подбора, основывающегося на векторной модели, выявило его недостатки, приводящие к некорректным результатам:

➤ Числовые характеристики номенклатуры (такие, как например, размер), обязанные находиться в словаре вектора, подразумевают бесконечное расширение его размера

➤ Редко используемый размер не попадет в вектор, а значит при подборе размер не будет учтен, что ухудшает эффективность подбора

Исходя из вышеописанной проблемы, алгоритм автоматического подбора был реализован на основе рейтингования.

В итоговом разработанном алгоритме автоматического подбора ранжирование представлено двумя видами:

- Исторический фактор

- Лингвистический подбор

При ранжировании осуществляется процедура предобработки:

1. Разбиение запроса от клиентской конфигурации 1С по разделителю «|» для отделения ID клиента от входящего содержания А (к которому необходимо подобрать номенклатуру)
2. Разбиение строки файла выгрузки «Содержания.csv» или «Номенклатура.csv» по разделителю-табуляции, получая строку В (содержание выгрузки или рабочее название номенклатуры соответственно)
3. Символы «-» «'» в строках А, В заменяются на « - » « ' »
4. Строки А, В разбиваются по пробелу и спецсимволам («/», «,», «.», «\») для получения массивов слов
5. Полученные слова из строк А, В подвергаются стеммизации (приведение к канонической форме, отсекая окончания и суффиксы русскоязычных слов)
6. Строки А, В могут быть вновь получены путем соединения слов в пункте 5 используя разделитель-пробел.

Исторический фактор задействует файл выгрузки «Содержания.csv» и оценивает соответствие существующих «исторических прецедентов» входящему содержанию А.

Релевантность рассчитывается при совпадении строк $A=B$. Изначальное значение релевантности строки выгрузки равняется 0. Далее значение релевантности рассчитывается как сумма значений двух факторов (формула 1):

$$R(A, B) = a + b \quad (1), \text{ где:}$$

- $a = [0; 1]$ – фактор совпадения ID клиентов (при совпадении $a=1$)
- $b = [0; 0,5]$ – фактор утверждения (при значении поля «Утвержден» $b=0,5$)

ID клиента и значение поля «Утвержден» получены разбиением на втором этапе предобработки. Таким образом, релевантность исторического

фактора увеличивается при совпадении клиентов и факте утверждения в строке выгрузки.

Лингвистический подбор применяется к строкам файла выгрузки «Номенклатура.csv» и оценивает соответствие существующих в «истории» рабочих названий номенклатуры В входящему содержанию А.

Изначальное значение релевантности строки выгрузки равняется 0. Далее значение релевантности рассчитывается по формуле 2:

$$R(A, B) = \sum_{i=0}^n f(A, b_i) + sizeFactor(A, B) \quad (2)$$

$$f(A, b_i) = \begin{cases} 1, & \text{если } b_i \in A \\ -1, & \text{если } b_i \notin A \end{cases}$$

где $f(A, b_i)$ – функция, результатом которой является 1 при наличии слова (в канонической форме) из В во входящей строке А и -1 при отсутствии. Данная функция позволяет повысить релевантность строки, в которой присутствуют слова, которые также имеются во входной строке, и понизить релевантность, если же такое слово во входной строке отсутствует. Кроме того, релевантность может быть увеличена функцией $sizeFactor$:

$$sizeFactor(A, B) = \frac{1}{\sum_{i=0}^n |x_n - y_n|} \quad (3)$$

где x, y – измерения вида $W \times H, W \times H \times D$, n – номер измерения – фактор размера, позволяющий сравнить числовые характеристики размеров типа допуская вещественный тип чисел W, H, D . Рассчитывается при наличии подстроки-характеристики во входящем содержании и в содержании строки выгрузки при совпадении числа измерений (Например, «10x20» и «12x24»). Нахождение характеристики размера в А и В осуществляется с использованием регулярного выражения, приведенного в листинге 2.

Листинг 2. Регулярное выражения для поиска характеристик размера

```
\d{[0-9]*[\0|\S]*(\.|\,)*[0-9]*[\0|\S]([\*|\*|\*][\0|\S]*[0-9]*(\.|\,)*[0-9]*)\{0,4\}\d
```

Данные виды ранжирования применяются в функциях стандартного и расширенного подбора. Лингвистический подбор применяется в случае невыполнения подбора по историческому фактору.

ГЛАВА 2. ОБЕСПЕЧЕНИЕ НАДЕЖНОСТИ И БЕЗОПАСНОСТИ СИСТЕМЫ

ПК-хост представляет собой виртуальную машину в собственности организации с установленной ОС Windows 10. ПК не имеет выхода в сеть Интернет, но доступен ко взаимодействию с компьютерами во внутренней локальной сети организации.

Существующее файловое хранилище организации расположено во внутренней сети, доступ к которой ограничен по сети Wi-Fi. Доступ к файлам хранилища регулируется системным администратором организации на уровне доменных политик и открыт для ПК-хоста.

Система не ведет работу с личными, а также какими-либо коммерческими данными организации либо её клиентов. Все лица-клиенты в выгружаемом файле (структура в приложении 1) представлены в виде обезличенных идентификаторов системы 1С. Система также не производит отправку данных за пределы внутренней сети организации.

Для обеспечения надежности системы ввиду политики обновлений ОС Windows 10, был настроен планировщик задач Windows. Были созданы следующие задачи:

- Перезагрузка ПК
 - Тип: перезагрузка системы
 - Триггер: по времени (каждый день в 00:30)
 - Цель: перезагрузка ОС для установки необходимых обновлений в целях предотвращения выполнения их установки в рабочее время
 - Реализация: планировщик задач Windows
- Запуск системы автоматического подбора
 - Тип: запуск приложения
 - Триггер: по запуску ОС

- Цель: автоматический запуск системы автоматического подбора после перезагрузки, сопровождающийся копированием актуальных файлов выгрузки из внешнего хранилища.
- Реализация: планировщик задач Windows, запускающий выполнение пакетного файла «start.bat». Файл содержит инструкцию вызова главного скрипта системы на языке Python:

Листинг 3. Автоматический запуск системы с помощью bat-файла

```
C:\Users\1\AppData\Local\Programs\Python\python.exe  
C:\Users\1\PycharmProjects\ita\main.py
```

ГЛАВА 3. ПРОВЕДЕНИЕ ИСПЫТАНИЙ СИСТЕМЫ

3.1 Испытание информационного обмена между сервером 1С и внешним ПМ

Данное испытание проводилось на начальных этапах разработки системы и заключалось в проверке обмена данными между двумя модулями системы, разделенных как физически, так и с точки зрения архитектуры системы (рисунок 6).

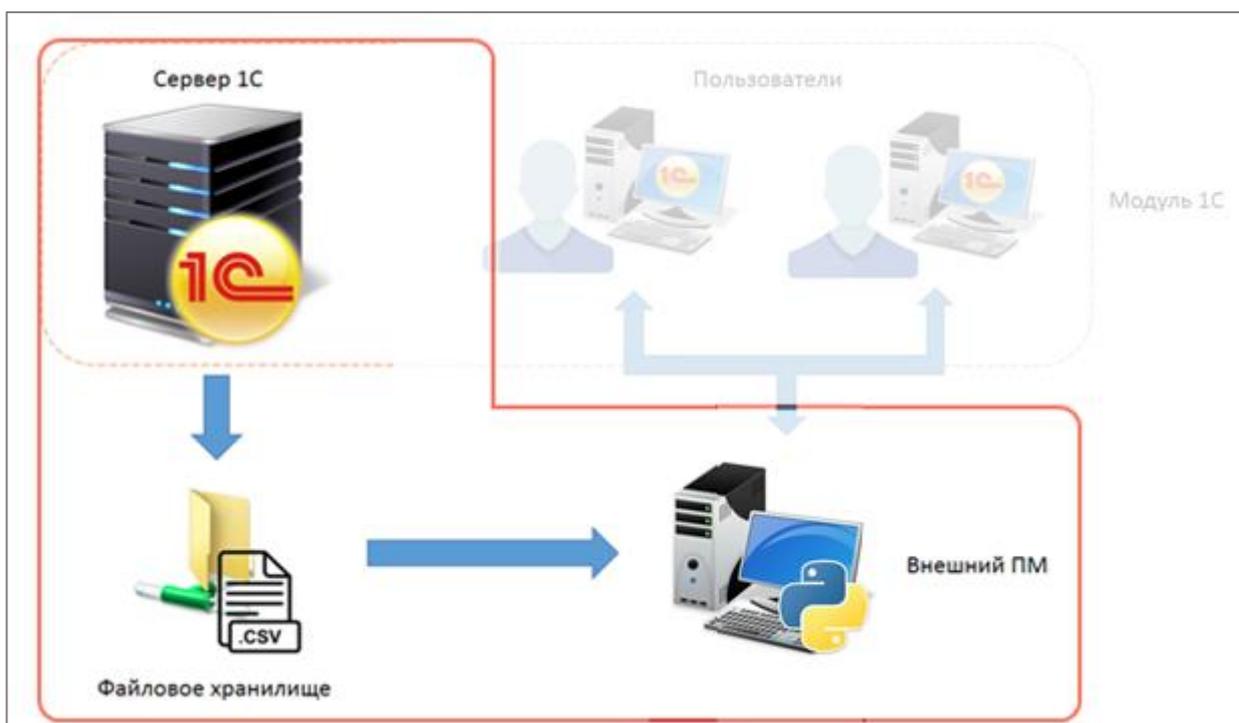


Рисунок 6. Схема модулей системы, задействованных в испытании №1

Целью испытания является получение внешним ПМ файла выгрузки содержаний КП и файла выгрузки номенклатуры, а также последующее открытие и считывание программой на языке Python.

Порядок испытания:

- Сервер 1С выгружает файл выгрузки содержаний КП содержаний (Содержания.csv) и файл выгрузки номенклатуры (Номенклатура.csv) в известную директорию файлового хранилища (\\server\public\data\ita\)
- Внешний ПМ копирует файлы (\\server\public\data\ita\Содержания.csv и <\\server\public\data\ita\Номенклатура.csv>) в папку проекта

- Внешний ПМ открывает вышеуказанные файлы, считывает и выводит строки в консоль программы
- Разработчик проверяет корректное отображение русскоязычных символов в выведенных строках

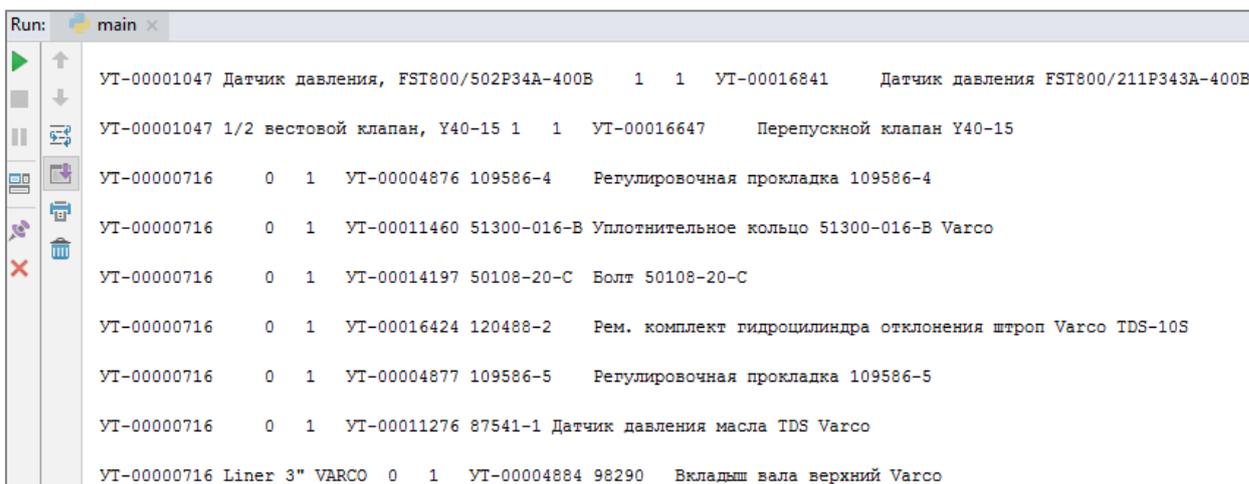
Для проведения испытания были реализованы:

- Регламентная выгрузка в конфигурации сервера 1С в соответствии со структурой данных в приложении 1.1 и 1.2
 - Задание выгрузки производится по расписанию (каждый день в 00:00, нерабочее время)
- С помощью программы на языке Python (листинг 4) производится получение файлов и их вывод

Листинг 4. Получение данных выгрузок внешним ПМ

```
import shutil
import codecs
Nom_file=None
Ord_file=None
...
shutil.copy2(r"\\server1-01\public\Data\NeuroNet\Выгрузка.txt", "Выгрузка.txt")
shutil.copy2(r"\\server1-01\public\Data\NeuroNet\Номенклатура_тест.txt",
"Номенклатура_тест.txt")
Nom_file = codecs.open('Номенклатура.txt', 'rU', encoding='UTF-8').readlines()
for row in Nom_file:
    print(row)
Ord_file = codecs.open('Содержания.txt', 'rU', encoding='UTF-8').readlines()
print(Ord_file)
for row in Ord_file:
    print(row)
```

В результате испытания были выведены строки выгружаемых файлов, с корректными русскоязычными символами в кодировке UTF-8 (рисунок 7). Испытание не превышает 7 часов, отведенных на обучающие и прочие нужды системы режимом обучения и заняло 6 секунд.



```
Run: main x
UT-00001047 Датчик давления, FST800/502P34A-400B 1 1 UT-00016841 Датчик давления FST800/211P343A-400B
UT-00001047 1/2 вестовой клапан, Y40-15 1 1 UT-00016647 Перепускной клапан Y40-15
UT-00000716 0 1 UT-00004876 109586-4 Регулировочная прокладка 109586-4
UT-00000716 0 1 UT-00011460 51300-016-B Уплотнительное кольцо 51300-016-B Varco
UT-00000716 0 1 UT-00014197 50108-20-C Болт 50108-20-C
UT-00000716 0 1 UT-00016424 120488-2 Рем. комплект гидроцилиндра отклонения штроп Varco TDS-10S
UT-00000716 0 1 UT-00004877 109586-5 Регулировочная прокладка 109586-5
UT-00000716 0 1 UT-00011276 87541-1 Датчик давления масла TDS Varco
UT-00000716 Liner 3" VARCO 0 1 UT-00004884 98290 Вкладыш вала верхний Varco
```

Рисунок 7. Консоль программы с выведенными строками файлов выгрузки

3.2 Испытание автоматического подбора и заполнения номенклатуры

Данное испытание проводилось при внедрении системы в процесс работы специалистов отдела продаж при подборе номенклатуры.

Целью испытания является проверка выполнения и вывода результатов подбора в интерфейсе пользовательской конфигурации 1С (выполнение стандартного и расширенного подбора), релевантность выведенных значений, а также оценка частоты использования (одобрения) специалистами результатов подбора от системы.

Порядок испытания:

- Разработчик проверяет работоспособность и релевантность подборов из пользовательской конфигурации 1С
- Специалисты ознакомятся с инструкцией пользователя (п. 5)
- Специалисты используют систему при подборе номенклатуры, работая над документами КП в течение недели
- Определяется процентная доля прецедентов, при которых специалист «одобрил» подбор, предоставленный системой и провел документ с данной номенклатурой

В ходе испытания разработанного функционала в пользовательской конфигурации и в программном модуле было отмечено соблюдение

релевантности выводимой номенклатуры запрашиваемому содержанию: так, окно расширенного подбора (рисунок 8) демонстрирует список наиболее подходящей номенклатуры для содержания «Втулку 100мм F800/1000». Верхние позиции при подборе занимает номенклатура, название которой практически идентично содержанию, ниже расположены менее релевантные единицы номенклатуры, названия которых содержат все меньше подстрок (слов) из содержания. Кроме того, при наличии полного совпадения по содержанию и/или совпадению клиента, участвовавшего в документе КП ранее, приоритет такой номенклатуры будет выше. Также, стандартный автоматический подбор заполняет позиции номенклатуры соответственно множеству содержаний в среднем за 5 сек (для 30 строк в КП в среднем).

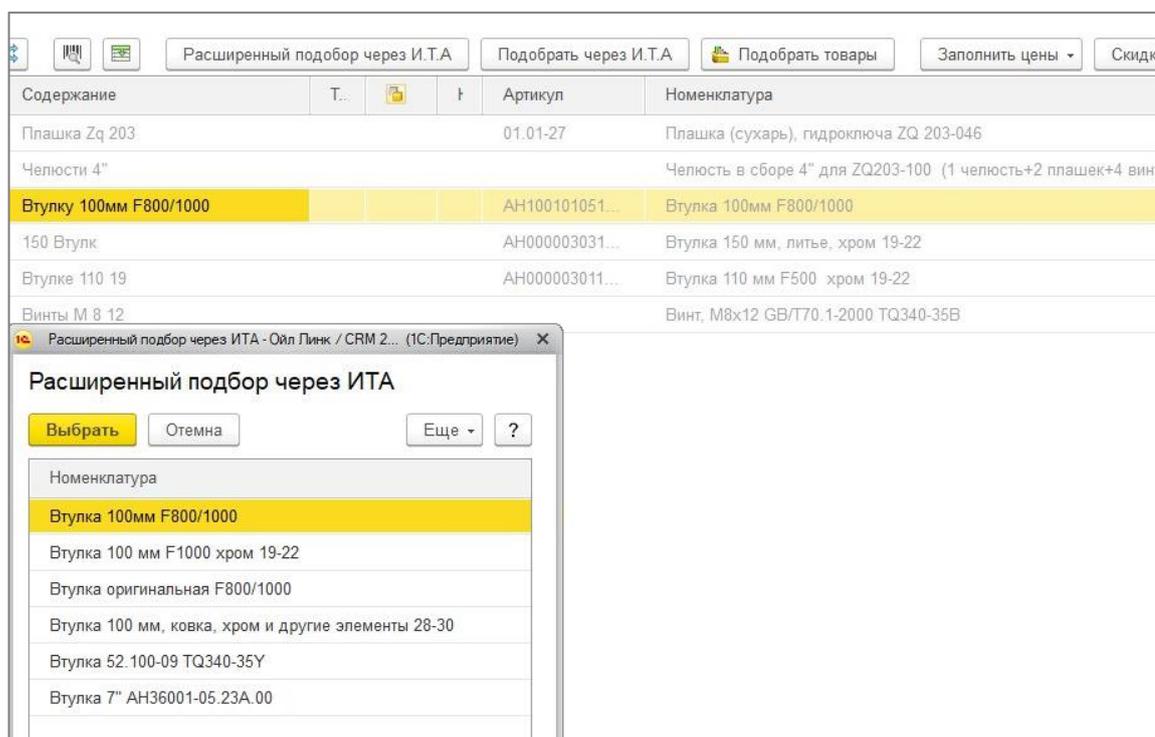


Рисунок 8. Интерфейс пользовательской конфигурации 1С при автоматическом подборе системой

В ходе испытания система использовалась специалистами отдела продаж в течение недели (20 рабочих дней). Так, специалист в среднем тратит 1,5 минуты на заполнение номенклатуры по содержанию (30 строк в среднем на документ КП), используя справочник «Номенклатура», а также обращаясь к собственным экспертным знаниям и опыту.

В табличную часть КП было внедрено числовое поле «Система» для отражения, применил ли специалист результат подбора системы. При обращении к подбору через систему и проведении документа КП, в каждой строке значение поля «Система» заполнялось «1». Если же специалист изменил номенклатуру после подбора, значения поля «Система» заполняется «2».

Была проведена выгрузка в файл формата «.xlsx» (отчёт) по позициям табличных частей КП, заполнявшимся специалистами в течение 7 рабочих дней с применением результатов системы или без него (рисунок 9).

1	2	3	4	5	6	7	8
Сальник, 190x240x18	Сальник В190x225x18	1	Коммерческое предложение клиенту 00УТ-				
Подшипник 22236СС/W33	Подшипник 22236СС/W33	1	Коммерческое предложение клиенту 00УТ-				
Сальник ВАУМХ7, 160x190x15	Сальник 160-190-15 ВАУМХ7	1	Коммерческое предложение клиенту 00УТ-				
Сальник ВАУМХ7, 145x175x15	Сальник 145-175-15 ВАУМХ7	1	Коммерческое предложение клиенту 00УТ-				
Подшипник 352230Х2 GB/Т 299-1	Подшипник 352230Х2 GB/Т 299-1995	1	Коммерческое предложение клиенту 00УТ-				
Подшипник 352224Е GB/Т299-1995	Подшипник 352224Е GB/Т 299-1995	1	Коммерческое предложение клиенту 00УТ-				
Сальник ВАУМSLX7, 130x160x12	Сальник 130-160-12 ВАУМSLX7	1	Коммерческое предложение клиенту 00УТ-				
Сальник ВАУМSLX7, 145x175x15	Сальник 145-175-15 ВАУМSLX7	1	Коммерческое предложение клиенту 00УТ-				
Сальник ВАУМSLX7, 120x150x12	Сальник 120-150-12 ВАУМSLX7	1	Коммерческое предложение клиенту 00УТ-				
Сальник ВАУМSLX7 105x130x12	Сальник 105-130-12 ВАУМSLX7	1	Коммерческое предложение клиенту 00УТ-				
Подшипник 30322 GB/Т297-1994	Подшипник 30322 GB/Т297-1994	1	Коммерческое предложение клиенту 00УТ-				
Подшипник 6019 GB/Т276-1994	Подшипник 6019 GB/Т276-1994	1	Коммерческое предложение клиенту 00УТ-				
Подшипник 61952 GB/Т276-1994	Подшипник 61952 GB/Т276-1994	1	Коммерческое предложение клиенту 00УТ-				

Рисунок 9. Внешний вид отчета использования системы специалистами

Вышеописанный отчет содержит:

- 648 прецедентов (строка табличной части КП с содержанием и соотв. номенклатурой и т.д.)
- 469 прецедентов имеют значение «1» поля «Система»
- 179 прецедентов имеют значение «2» поля «Система»

Таким образом, по итогам испытания можно сделать вывод: в течение 20 рабочих дней специалисты использовали функционал системы и применили (сохранили) результат работы системы в 72% случаев. Кроме того, в течение данного периода специалисты сократили свои временные затраты в среднем до 2,14 раз (0.7 мин вместо 1,5 мин), тем самым повышая собственную эффективность операционной деятельности.

ГЛАВА 4. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Руководство пользователя распространяется среди специалистов отдела продаж в организации в физическом виде на бумажном носителе и/или электронного документа в формате «.pdf». Руководство даёт следующую инструкцию по применению системы в целях осуществления автоматического подбора номенклатуры документа КП в пользовательской конфигурации 1С:

1. Находясь в окне списка документов КП, откройте существующий документ КП либо нажмите «Создать» для заполнения нового.

2. Находясь в окне редактирования документа КП, выберите клиента в поле «Клиент» (это действие не обязательно для выполнения автоматического подбора, но при заполнении может ориентировать подбор на предыдущие заказы данного клиента). Перейдите во вкладку «Товары» для отображения табличной части для ввода содержаний и соответствующей номенклатуры.

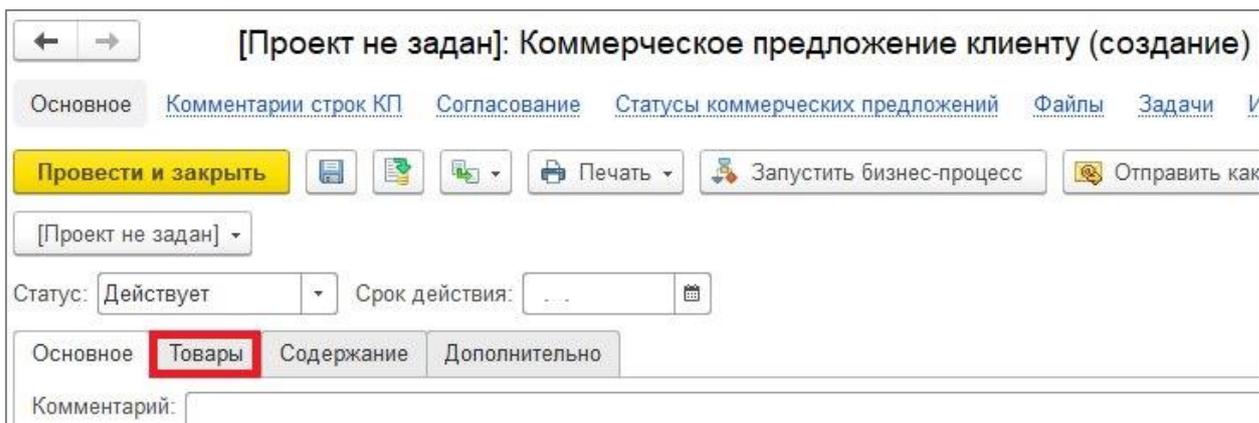


Рисунок 10. Окно документа КП в руководстве пользователя

3. Находясь во вкладке «Товар», заполните табличную часть, добавляя строки нажатием кнопки «Добавить». Для последующего запуска автоматического подбора требуется заполнить только поле «Содержание» на естественном языке. Вы можете заполнить от 1 до множества строк.

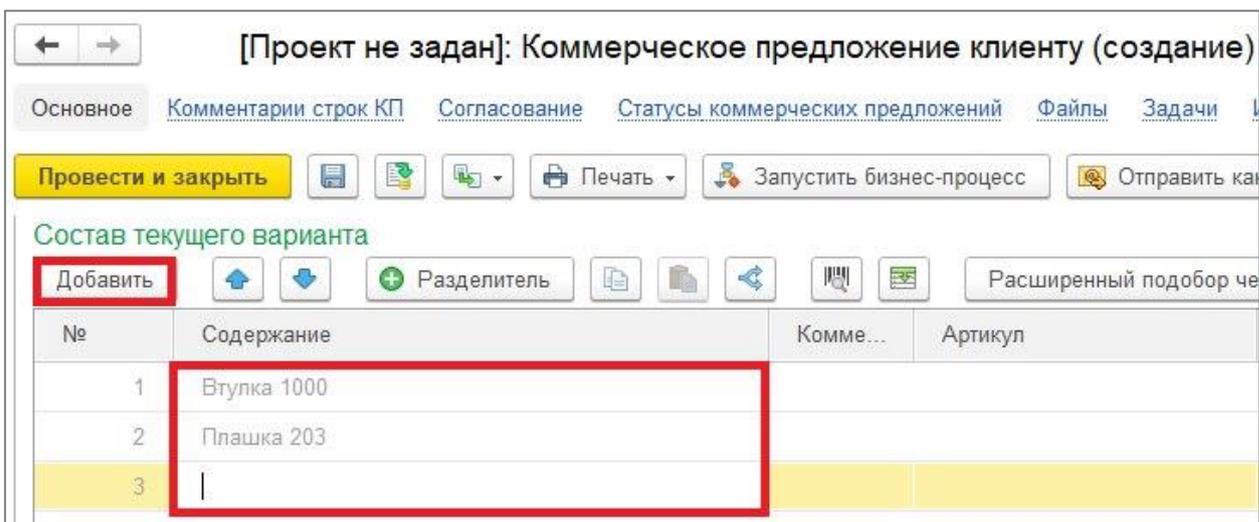


Рисунок 11. Окно вкладки «Товары» КП в руководстве пользователя

4. Для запуска автоматического подбора номенклатуры нажмите кнопку «Подобрать через И.Т.А». Система заполнит поле «Номенклатура» элементами, наиболее подходящими относительно введенных содержаний. Система выбирает ту номенклатуру, рабочее название которой наиболее совпадает с содержанием (при этом окончания слов не учитываются). Система также учитывает все существующие проведенные документы КП с их содержаниями и номенклатурой. Если идентичному содержанию ранее уже подбиралась номенклатура, то она будет использована в качестве результата (исторический фактор подбора). Если «в истории» существует несколько идентичных содержаний, то система поставит в приоритет строки с совпадением клиента (поле «Клиент» документа КП, к которому относится табличная часть с данной строкой). Также приоритет в меньшей степени повышает факт утверждения варианта.

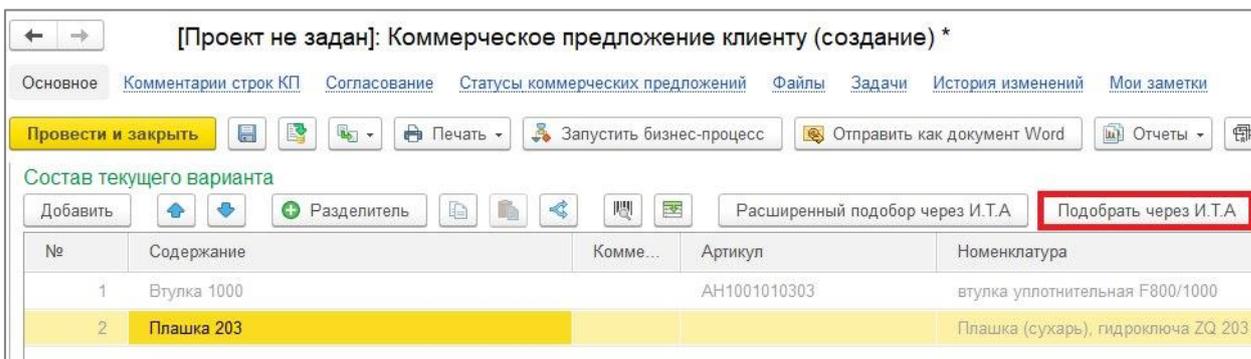


Рисунок 12. Окно вкладки «Товары» после нажатия «Подобрать через И.Т.А»

5. Автоматический подбор также может выполняться в расширенном режиме, позволяющим предоставлять сразу несколько подходящих вариантов номенклатуры на выбор. Для его запуска, нажмите «Расширенный подбор через И.Т.А.», выбрав одну из строк табличной части с заполненным содержанием. Вам будет выведено окно, содержащее список наиболее подходящей номенклатуры в порядке убывания и кнопками «Выбрать» и «Отмена». Выберите нужную строку в списке и нажмите «Выбрать», после чего номенклатура в табличной части документа КП заполнится выбранной вами в расширенном подборе.

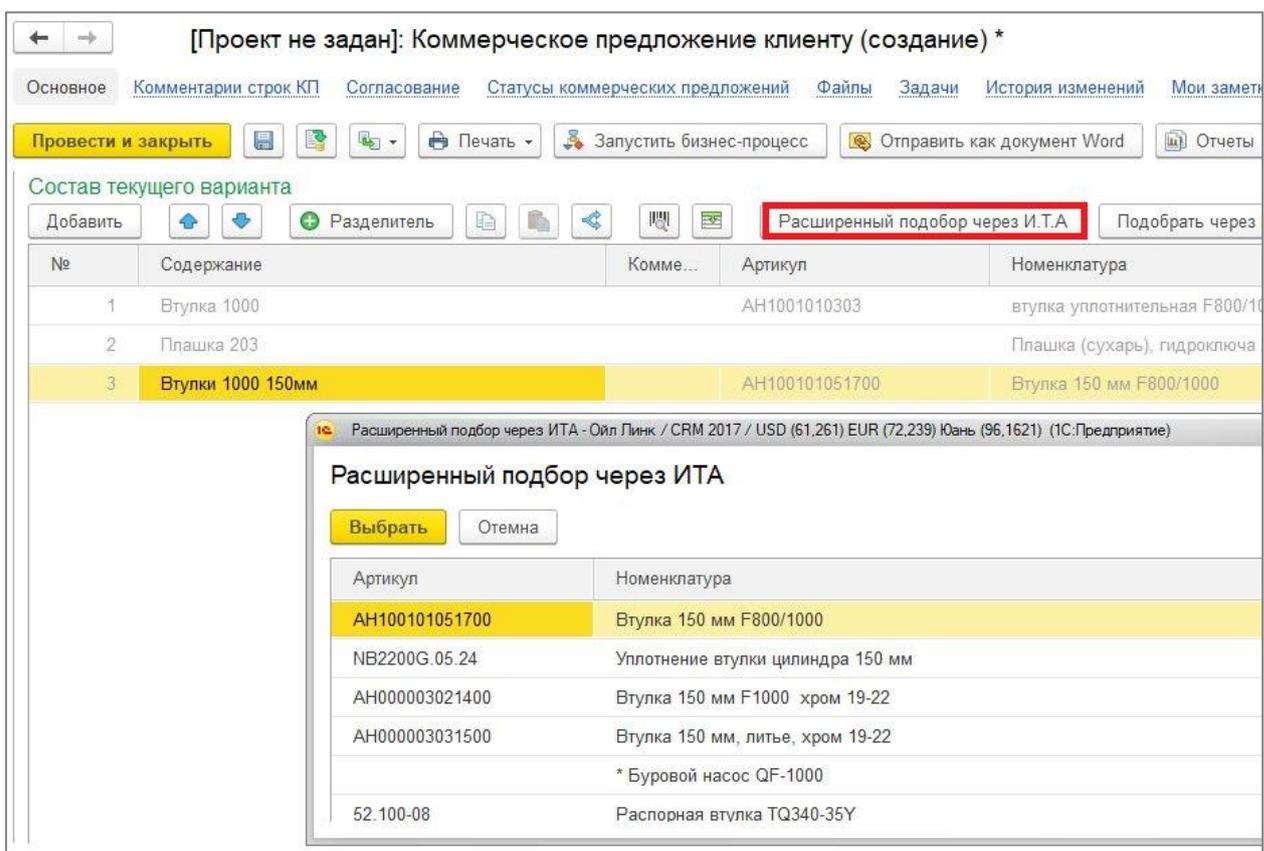


Рисунок 13. Расширенный подбор через И.Т.А

ГЛАВА 5. ОЦЕНКА ОРГАНИЗАЦИОННОЙ ЭФФЕКТИВНОСТИ ПП

Оценка организационной эффективности программного продукта (по методическим указаниям Ивахника Д.Е. [11]) проводилась с принятием во внимание следующих данных:

- Среднее время заполнения (подбора номенклатуры) КП = среднее время заполнение 1 строки (номенклатуры) * среднее число строк табличной части «Товары» КП = 1,5 мин * 30 строк = 45 мин./КП
- Среднее время автоматического подбора = среднее время обработки системой + среднее время перепроверки строк специалистом = (5 сек + 35 сек) * среднее число строк = 0,67 мин. * 30 строк = 20 мин./КП

Таблица 1. Оценка организационной эффективности ПП

Показатели организ. эффективности и	Вес показателя	Значение показателя до внедрения ПП, баллов	Значение показателя до внедрения ПП с учетом веса (гр. 2 * гр. 3)	Значение показателя после внедрения ПП, баллов	Значение показателя после внедрения ПП с учетом веса (гр. 2 * гр 5)
1	2	3	4	5	6
Среднее время подбора номенклатуры одного КП	1	45 минут	1 * 45 = 45 минут	20 минут	1 * 20 = 20 минут

Результирующее значение роста организационной эффективности разработки ПП осуществляется путем соотнесения итоговых значений гр. 6 и гр. 4 вышеуказанной таблицы. Таким образом, высвобождается и перенаправляется на другие задачи $(1 - (20/45)) * 100\% = 55,56\%$ от трудовых затрат специалистов на определение и подбор номенклатуры.

Таким образом, с учетом среднего количества КП в месяц (96), среднего затрачиваемого времени на КП (45 мин) и освободившегося времени на каждом КП (45-20=25мин), высвобождается количество человеко-часов, равное: $25 * 96 / 60 = 40$ человеко-часов.

ЗАКЛЮЧЕНИЕ

В ходе работы были выполнены поставленные цели и задачи, приведенные ниже.

Разработано программное обеспечение:

- Определен функционал и архитектура системы
- Произведены доработки серверной и пользовательской

конфигураций 1С:Предприятие (УТ)

- Реализован веб-сервер на языке Python

Разработан механизм ранжирования для автоматического подбора:

- Разработан алгоритм подбора номенклатуры на языке Python в двух вариациях (стандартный и расширенный) на основе механизма рейтингования

Обеспечена безопасность и надежность системы:

- Операции системы реализованы во внутренней сети организации
- Настроен планировщик задач Windows для предотвращения отключения программы автоматического подбора из-за непредвиденных перезагрузок ОС

Проведены испытания системы:

- Разработан механизм выгрузки данных на файловое хранилище и их получение внешним программным модулем

• Разработан механизм отслеживания применения результатов подборов системы специалистами отдела продаж и выполнена предварительная оценка данного применения

- Система введена в предварительную эксплуатацию специалистами отдела продаж

Написано руководство пользователя для работы с системой;

Проведена оценка организационной эффективности ПП.

СПИСОК ЛИТЕРАТУРЫ

1. Википедия - Ранжирование [Электронный ресурс] URL <https://ru.wikipedia.org/wiki/Ранжирование> (дата обращения: 20.02.18)
2. Релевантность и ранжирование — что это такое и какие факторы влияют на положение сайтов в выдаче Яндекса и Гугла [Электронный ресурс] URL: <http://ktonanovenkogo.ru/seo/search/relevantnost-ranzhirovanie-chto-eto-takoe.html> (дата обращения: 20.02.18)
3. scikit learn - Naive Bayes [Электронный ресурс] URL: http://scikit-learn.org/stable/modules/naive_bayes.html (дата обращения: 05.10.17)
4. Machine Learning Mastery - How To Implement Naive Bayes From Scratch in Python [Электронный ресурс] URL: <https://machinelearningmastery.com/naive-bayes-classifier-scratch-python/> (дата обращения: 05.10.17)
5. Python 3 для начинающих - Язык программирования Python 3 [Электронный ресурс] URL: <https://pythonworld.ru/> (дата обращения: 16.09.17)
6. Дмитрий Акинин - PyStemmer — стемминг в Python [Электронный ресурс] URL: <http://akinin.info/post/52041640243/pystemmer-стемминг-в-python> (дата обращения: 18.09.17)
7. 1С:Предприятие 8 – Регламентное задание [Электронный ресурс] URL: http://v8.1c.ru/overview/Term_000000154.htm (дата обращения: 09.10.17)
8. Википедия – Интеллектуальная система [Электронный ресурс] URL: https://ru.wikipedia.org/wiki/Интеллектуальная_система (дата обращения: 05.09.17)
9. Ойл Линк [Электронный ресурс] URL: <http://oillink.ru/> (дата обращения: 05.09.17)
10. Миллер Ю.О., Бидуля Ю.В. РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОГО АССИСТЕНТА СПЕЦИАЛИСТА ОТДЕЛА ПРОДАЖ // Математика и междисциплинарные исследования – 2018 [Электронный ресурс]: материалы Всерос. науч.-практ. конф. молодых ученых с междунар. участием (г. Пермь, 14–19 мая 2018 г.) / гл. ред. А. П.

Шкарапута; Перм. гос. нац. исслед. ун-т. – Электрон. дан. – Пермь, 2018. – с. 68-71.

11. Предварительная оценка организационно-экономической эффективности разработки программного обеспечения: методические указания /Сост.: Д. Е. Ивахник

Приложение 1.

В данном приложении приводятся структуры данных файлов выгрузок, а также структуры данных, использующиеся непосредственно алгоритмом автоматического подбора.

1.1 Структура данных файла «Содержания.csv» при выгрузке сервером 1С:

- ID Клиента (текст, обозначающий клиента, чье содержание заказа сопоставляется с номенклатурой). Источник: «Клиент» документа КП
- Содержание заказа (текстовые данные на естественном языке, которым оригинально вручную специалистом сопоставляется номенклатура).

Источник: элемент «Содержание» строки в табличной части «Состав текущего варианта» документа КП

- ID номенклатуры (текст, обозначающий сопоставляемую номенклатуру). Источник: элемент «Номер» строки в табличной части «Состав текущего варианта» документа КП

- Утвержден (флаг 0 или 1, означающий утвержден ли документ КП, к которому относится строка (прецедент) табличной части с содержанием, номенклатурой и т.д. Источник: флаг «Утвержден» документа КП

1.2 Структура данных файла «Номенклатура.csv» при выгрузке сервером 1С:

- ID номенклатуры (текст, обозначающий сопоставляемую номенклатуру). Источник: поле «Номер» элемента в справочнике «Номенклатура»

- Рабочее название номенклатуры (текст). Источник: одноименное поле элемента в справочнике «Номенклатура»

1.3 Структура данных, поступающих от пользователей 1С:Предприятие (УТ) алгоритму автоматического подбора внешнего ПМ:

- ID Клиента (текст, обозначающий клиента, чье содержание заказа сопоставляется с номенклатурой)

- Содержание заказа (1 для расширенного подбора и более 1 элементов для стандартного в зависимости от числа позиций) – текстовые данные на естественном языке, которым оригинально вручную специалистом сопоставляется номенклатура

1.4 Структура данных, поступающих от алгоритма автоматического подбора внешнего ПМ пользователям 1С:

- ID номенклатуры (1 и более элементов) – текст, обозначающий сопоставляемую номенклатуру

ТЕХНИЧЕСКОЕ ЗАДАНИЕ
на разработку подсистемы интеллектуального ассистента специалиста
«И.Т.А.»

Заказчик:
ООО «Ойл Линк»
Епифанов В. С.

Исполнитель:
Миллер Ю. О.

2017 г.

Аннотация

В научно-исследовательской работе приводится написанный в ходе работы документ, являющийся техническим заданием на разработку системы, обеспечивающей анализ, хранение и обмен данных между существующей базой данных организации и внешнего разрабатываемого программного модуля интеллектуального анализа с целью реализации функций интеллектуального ассистента пользователя системы 1С:Предприятие (УТ).

В данном документе определяются общие сведения о системе, описание составляющих частей системы, включая её структуру. Приводятся требования к надёжности и безопасности, к обслуживающему персоналу системы, к информационному и техническому обеспечению, а также требования к документированию. Приводится состав и содержание работ, порядок контроля и приемки системы, а также и источники разработки.

Общие сведения

Полное наименование системы – интеллектуальный ассистент специалиста отдела продаж

Краткое наименование системы – интеллектуальный ассистент («И.Т.А.»).

Цель внедрения системы: высвобождение до 40 человеко-часов в месяц (среди специалистов отдела продаж) за счёт автоматизации функции подбора номенклатуры.

Определения, обозначения и сокращения

- Система – аналитическая система в целом; эквивалентно полному наименованию системы «интеллектуальный ассистент специалиста отдела продаж».

- Пользователь – специалист отдела продаж и клиент 1С:Предприятие (УТ), осуществляющий подбор номенклатуры

- Клиент – конфигурация 1С

- УТ – управление торговлей

- КП – коммерческое предложение клиенту организации

- Заказ – соответствие содержания, полученного от определенного клиента организации, заданной номенклатуре

- Содержание – позиция номенклатуры в заказе, сформулированная клиентом организации в естественном виде и задействованная в документе КП

- ПО – программное обеспечение

- ПМ – программный модуль

- ИО – информационный обмен

- ID – Уникальный идентификатор

- API – Application Programming Interface – программный интерфейс приложения

1. Описание основных частей системы

Среди основных частей системы выделены функциональная и обеспечивающая части.

1.1. Функциональная часть системы

Функцией системы является подбор номенклатуры для содержания заказа документа КП в интерфейсе клиента 1С:Предприятие (УТ). Подбор основан на анализе ранее заполненных КП (т.е. существуют товарные позиции с заполненными полями «Содержание» и «Номенклатура»). При этом нужно понимать, что в нормальном режиме система функционирует следующим образом:

1) Пользователь (специалист отдела продаж) при добавлении или редактировании документа КП заполняет клиента и содержания заказа (рисунок 1) в табличной части вкладки «Товары» (товарные позиции на естественном языке). Затем, пользователь нажимает на кнопку «Подобрать через И.Т.А.».

№	Содержание	Комментарий	Артикул	Номенклатура
1	Счетчик *модуля			
2	Соединение ZQ203			

Рисунок 1. Интерфейс клиента конфигурации 1С:Предприятие (УТ)

2) Система получает значение ID (идентификатора) клиента на основании заполненного пользователем поля «Клиент» и значения позиций

колонки «Содержание» из табличной части вкладки «Товары» в следующем виде. Полученные строки в единую строку с разделителями.

3) Система на стороне клиента конфигурации 1С:Предприятие (УТ) производит GET-запрос по соответствующему адресу:

➤ «http://10.1.124.30:8080/УТ-0000092|Счетчик *модуля|Соединение ZQ203», где 10.1.124.30:8080 – адрес веб-сервиса внешнего ПМ в локальной сети организации, остальное за чертой «/» - строка, полученная в пункте 2

4) Система принимает строку благодаря произведенному запросу на веб-сервис и разбивает её на подстроки ID (идентификатор) клиента и одно или несколько содержаний. В каждой такой подстроке-содержании, например, «Счетчик *модуля», удаляются все символы, кроме букв и цифр, а также устанавливается нижний регистр. Кроме того, слова в каждой подстроке-содержании (строки по пробелам) подвергаются отсечению окончаний.

5) Каждая подстрока-содержание, полученная вследствие преобразований выше, преобразуется в векторный вид (ряд из 300 чисел 1 или 0) на основании наличия или отсутствия в строке каждого из 300 элементов словаря. Формирование словаря описано ниже в рамках режима обучения. К каждому полученному ряду чисел ставится в пару строка, представляющая ID (идентификатор) клиента. Так, если в словаре первые слова: «счетчик», «соединен», «кольцо», «модуль», «zq203» ..., пары ID и содержаний примут следующее представление:

➤ «УТ-0000092», 1 0 0 1 0 ... (строка и ряд чисел)

➤ «УТ-0000092», 0 1 0 0 1 ... (строка ряд чисел)

6) Алгоритм, обученный на выборке соответствий пары [ID Клиента (строка), содержание (ряд чисел)] идентификатору ID номенклатуры (строка), подбирает наиболее подходящий идентификатор номенклатуры каждой паре. Для подбора номенклатуры применяется метод обучения на предыдущих заказах.

➤ «УТ-00007079» (строка)

➤ «УТ-00004389» (строка)

7) Полученные строки, обозначающие ID (идентификатор) подобранной номенклатуры объединяются в единую строку, которая возвращается клиенту конфигурации 1С:Предприятие (УТ) в качестве строкового ответа на запрос.

8) Система на стороне клиента конфигурации 1С:Предприятие (УТ) разбивает полученную выше строку на несколько строк, представляющих собой ID (идентификаторы) подобранной номенклатуры и заполняет значения колонки «Номенклатура» табличной части вкладки «Товары», подставляя номенклатуру по идентификатору из справочника «Номенклатура». Подстановка производится по порядку, сохраняя соответствие содержания номенклатуре (рисунок 2).

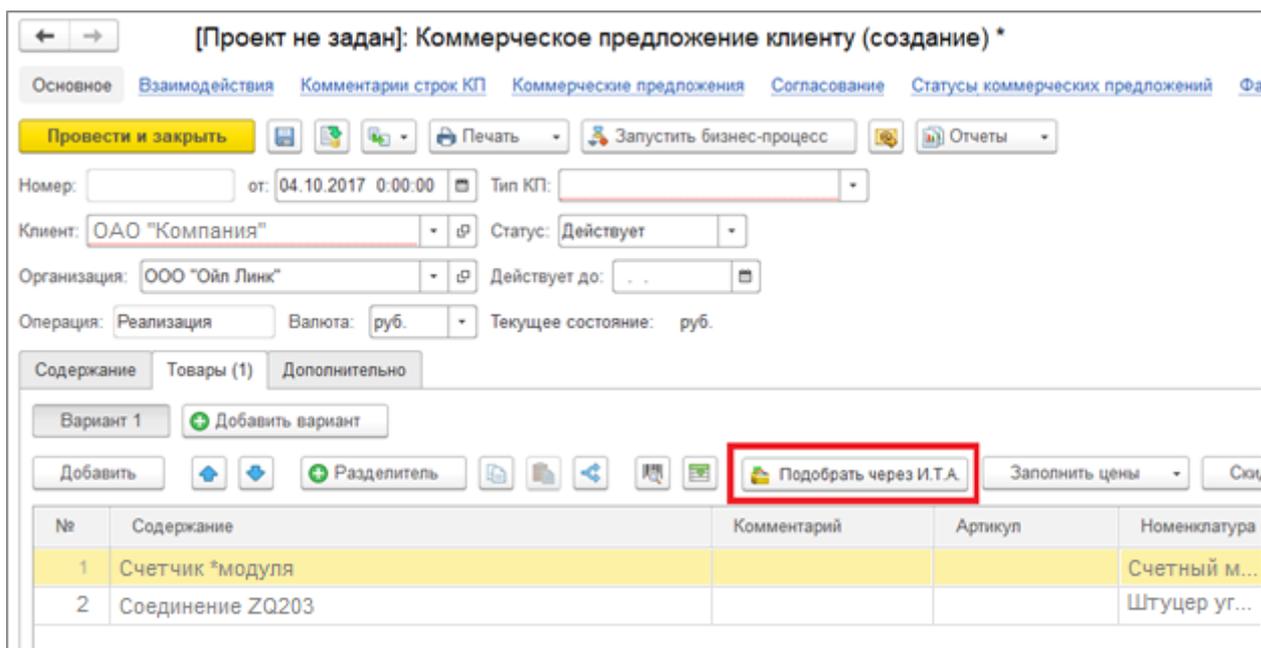


Рисунок 2. Интерфейс клиента 1С:Предприятие (УТ) в результате функционирования системы

Функции системы в режиме обучения можно описать следующей последовательностью действий:

1) Доработанная конфигурация сервера 1С:Предприятие (УТ) каждый день в определенное время формирует файл-выгрузку из БД, содержащую данные согласно структуре в приложении 1.1. Строки файла выглядят следующим образом («\t» обозначает символ табуляции):

- «УТ-0000092 \t Счетные модули \t УТ-00007079»
- «УТ-0000091 \t Штуцеры угловые \t УТ-00004389»
- ...

2) Сформированный файл «Выгрузка.csv», содержащий вышеописанные данные, сохраняется в файловом хранилище организации, заменяя одноименный ранее выгруженный файл при наличии такового

3) Система на стороне внешнего ПМ копирует файл пункта 2 в папку исполняемого проекта внешнего ПМ, заменяя существующий при наличии такового. Затем система формирует файл «Словарь.txt», представляющий собой список самых используемых слов в содержаниях (берутся 2-е элементы с учетом разделителя в строках выгрузки). Для этого, система удаляет из строк содержаний все символы, кроме букв и цифр, приводит строки к нижнему регистру, разбивает их по разделителю-пробелу и отсекает окончания слов

4) Для каждой уникальной строки подсчитывается частота по всей выгрузке. Так, строке «счетн» будет поставлено в пару число 50, если оно встретилось в выгрузке 50 раз. В итоговый словарь входят лишь 300 первых строк с наивысшей частотой. Итоговый файл «Словарь.txt» имеет построчный вид:

- «счетч»
- «соединен»
- ...

5) Система преобразовывает 2-й элемент по разделителю (содержание) каждой строки файла «Выгрузка.csv». Например, строка

«Счетные модули» подвергается удалению всех символов, кроме букв и цифр, приведению к нижнему регистру, разбиению по разделителю-пробелу и отсечению окончаний получившихся слов.

б) Каждое содержание вида [«счетн», «модул»] полученное вследствие преобразований выше, преобразуется в векторный вид (ряд из 300 чисел 1 или 0) на основании наличия или отсутствия в строке каждого из 300 элементов словаря аналогично преобразованиям в нормальном режиме функционирования, описанного выше. Итоговый файл «Выгрузка.csv» имеет следующий вид:

- «УТ-0000092 \t 00101 ... \t УТ-00007079»
- «УТ-0000091 \t 00000 ... \t УТ-00004389»
- ...

7) Алгоритм автоматического подбора системы на стороне внешнего ПМ обучается на основании «истории» соответствий в предобработанном выше файле «Выгрузка.csv». Обучение требуется для выявления наиболее подходящей номенклатуры (идентификатора) для пары [ID Клиента (строка), содержание (ряд чисел)].

Предполагаемая задача может быть рассмотрена как задача классификации с применением традиционных подходов. Так, признаками для алгоритма машинного обучения при классификации будут являться первые два элемента строк по разделителям – идентификатор клиента и содержание в виде ряда чисел, а классом – идентификатор номенклатуры в виде строки.

- «УТ-0000092», 00101 ... -> «УТ-00007079»
- «УТ-0000091», 00000 ... -> «УТ-00004389»
- ...

Система должна функционировать:

- В рабочее время, определяемое нормальным режимом функционирования (пункт 1.1.4)
- Автоматический подбор и заполнение номенклатуры не должно превышать 10 секунд, за исключением случаев при времени отклика (ping), превышающим 10 секунд, вызванным аппаратными и/или программными неисправностями компонентов, не входящими в состав системы

Пример ассоциации данных представлен в Приложении 2. Иллюстрация требуемого образа работы системы приведена на рисунках 1 и 2.

1.1.1. Состав модулей системы

- 1. Модуль 1С (Доработанная конфигурация 1С:Предприятие (УТ))
 - 1.1 Доработанная конфигурация клиента 1С:Предприятие (УТ) (пользователи)
 - 1.1.1 Доработанный интерфейс документа КП (добавление и редактирование)
 - 1.1.2 API для ИО с внешним ПМ
 - 1.2 Доработанная конфигурация сервера 1С:Предприятие (УТ) (Механизм выгрузки)
- 2. Внешний ПМ
 - 2.1 Механизм загрузки данных (файлового хранилища организации) от сервера 1С:Предприятие (УТ)
 - 2.2 Веб-сервис с API для ИО с клиентами 1С:Предприятие (УТ)
 - 2.3 Алгоритм автоматического подбора

Состав модулей, включая сервер 1С:Предприятие (УТ), пользователей, файловое хранилище организации (включая выгрузку данных в виде файла формата .csv) и внешний ПМ с API, проиллюстрирован на рисунке 3.

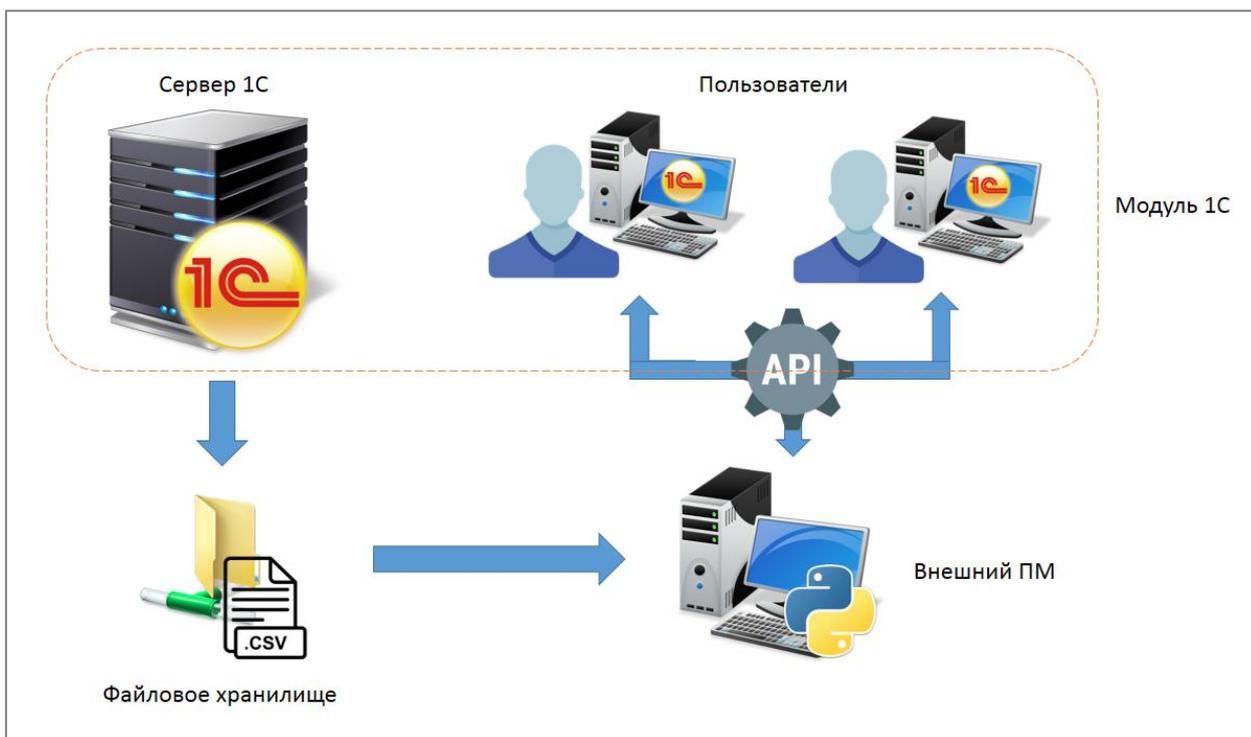


Рисунок 3. Схема модулей системы

1.1.2. Требования к функциям модулей системы

Требования к функциям модулей системы (внутренних подсистем), входящих в состав системы, приведены ниже:

- 1) Модуль 1С (Доработанная конфигурация сервера и клиента 1С:Предприятие (УТ))

Модуль должен предоставлять пользовательский интерфейс в конфигурации 1С:Предприятие (УТ) для вызова автоматического подбора и заполнения, а также проводить выгрузку данных для внешнего ПМ, согласно структуры данных в Приложении 1.1 и режиму обучения, описанному в пункте 1.1.4.

- 1.1) Доработанная конфигурация клиента 1С:Предприятие (УТ)

Доработанная конфигурация клиента 1С:Предприятие (УТ) включает в себе внесенные изменения в пользовательский интерфейс и сопутствующий API для отправки данных алгоритму внешнего ПМ согласно структуры в приложении 1.2 и прима и вывода ответных данных согласно приложению 1.3. Должна быть предусмотрена кнопка «Подобрать через И.Т.А.», находящаяся

над табличной частью документа КП, вызывающую процедуры API стороне клиента (рисунок 1).

При этом, должны учитываться заполненные поля «Клиент», а также позиции колонки «Содержание» в табличной части. Результат после работы алгоритма внешнего ПМ и передачи данных на сторону конфигурации клиента 1С:Предприятие (УТ) должен заключаться в заполненной колонке «Номенклатура» для позиций табличной части (рисунок 2).

1.2) Доработанная конфигурация сервера 1С:Предприятие (УТ)

Доработанная конфигурация сервера 1С:Предприятие (УТ) включает в себе изменения, внесенные в конфигурацию сервера 1С:Предприятие (УТ) внутри организации: должна осуществляться выгрузка данных в файловое хранилище организации согласно режиму обучения, описанному в пункте 1.1.4 и структуре данных в приложении 1.1.

2) Внешний ПМ

Внешний ПМ включает в себе механизм загрузки данных из файлового хранилища (структура в приложении 1.1), веб-сервиса с API для ИО с конфигурацией клиента 1С:Предприятие (УТ), а также реализованным алгоритмом автоматического подбора. Модуль должен поддерживать функционирование интеллектуального алгоритма подбора и производить получение его входных данных и отправку результата посредством функционирования API с учетом структур данных в приложениях 1.2 и 1.3. Алгоритм должен быть подкреплён математическим аппаратом, описанным в пункте 1.2.1 Автоматический подбор не должен превышать 10 секунд, за исключением случаев при времени отклика (ping), превышающим 10 секунд, вызванным аппаратными и/или программными неисправностями компонентов, не входящими в состав системы.

1.1.3. Способы ИО между модулями системы

Внутренние модули общей системы должны обмениваться данными посредством соединения по локальной сети внутри организации.

Доработанный клиент конфигурации 1С:Предприятие (УТ) должен обмениваться с внешним ПМ за счет API веб-сервиса, осуществляющим работу на ПК-хосте, и API на стороне клиентов 1С:Предприятие (УТ) (Передача данных для нужд алгоритма автоматического подбора согласно Приложению 1.2 и 1.3).

Особые требования к модулям, участвующим в ИО:

- Механизм выгрузки данных для внешнего ПМ должен производить выгрузку в отдельный каталог в файловом хранилище организации в локальной сети организации с учетом требований к безопасности, определенным в пункте 2.1. Выгрузка представляется файлом, заменяемым при каждом таковом прецеденте согласно режима обучения в пункте 1.1.4

1.1.4. Требования к режимам функционирования системы

Система должна обеспечивать работу главной функции (автоматический подбор) в любое время (нормальный режим), за исключением временного периода обучения алгоритма ИА в нерабочее время (режим обучения). Перечень режимов функционирования отражен в таблице 1.

Таблица 1. Перечень режимов функционирования системы

№	Режим	Описание	Временные рамки
1.	Нормальный режим	Доступный к работе автоматический подбор для пользователей	7:00 – 00:00 5 дней в неделю
2.	Режим обучения	Автоматическая подбор для пользователей недоступен	00:00 – 7:00 5 дней в неделю

		Обучение алгоритма автоматического подбора внешнего ПМ	
--	--	--	--

1.2. Обеспечивающая часть системы

К обеспечивающей части системы относятся математический аппарат и регламент работы пользователя. Кроме того, работа системы должна обеспечиваться средствами ОС Windows 7 или 10, а также средств языка Python или C#. Подробные требования к информационному обеспечению описаны в пункте 2.3.1.

1.2.1. Математический аппарат

Математический аппарат применяется в алгоритме автоматического подбора внешнего ПМ. В качестве математического аппарата системы выступают любые алгоритмы машинного обучения, справляющиеся с задачей классификации:

- Статистические методы классификации
 - Наивный Байесовский классификатор

Наивный байесовский классификатор (НБК) – это алгоритм классификации, основанный на теореме Байеса с допущением о независимости признаков. Другими словами, НБК предполагает, что наличие какого-либо признака в классе не связано с наличием какого-либо другого признака. Теорема Байеса позволяет рассчитать вероятность того, что конкретная причина привела к наблюдаемому событию зная, с какой вероятностью причина приводит к некоему событию. НБК позволяет проводить многоклассовую классификацию по нескольким признакам.
- Нейронные сети
 - Многослойный перцептрон

Многослойный перцептрон - искусственная нейронная сеть, состоящая из слоев, каждый из которых состоит из элементов —

нейронов. Модель нейрона - это элемент сети, который имеет несколько входов, каждый из которых имеет вес. Нейрон, получая сигнал, умножает сигналы на веса и суммирует получившиеся величины, после чего передает результат к другому нейрону или на выход сети. Обучение нейронных сетей данного типа проводится по алгоритму обратного распространения ошибки (ответ нейронной сети сравнивается с верным ответом и вычисляется ошибка, корректирующая веса предыдущих слоев. Как правило, нейронные сети оказываются наиболее эффективным способом многоклассовой классификации по множеству признаков, потому что генерируют фактически большое число регрессионных моделей (которые используются в решении задач классификации статистическими методами).

1.2.2. Регламент работы пользователя

Для получения «предложения» в виде подобранной на основе анализа алгоритмом автоматического подбора и автозаполнения, пользователь должен выполнить следующие действия в интерфейсе клиента 1С:Предприятие (УТ):

- Открыть окно добавления или редактирования документа КП
- Выбрать клиента в поле «Клиент»
- Перейти во вкладку «Вариант»
- Добавить новый вариант нажатием кнопки «Добавить вариант»
- Добавить 1 или более позиций номенклатуры в табличную часть нажатием кнопки «Добавить»
 - Заполнить значения ячеек колонки «Содержания» для всех позиций номенклатуры
 - Нажать кнопку «Подобрать через И.Т.А.»

Демонстрация пользовательского интерфейса до нажатия кнопки представлена на рисунке 1, после – на рисунке 2.

2. Требования к системе

Система, включая её модули, должна быть запущена в эксплуатацию в главном офисе организации на базе аппаратного и программного обеспечения, принадлежащему заказчику, указанному в п. 3.

Среди требований к системе также присутствуют требования к надежности и безопасности, требования к обслуживающему персоналу, требования к видам обеспечения, а также требования к документированию.

2.1. Требования к надежности и безопасности

Доступ и работа с данными, выгружаемыми сервером 1С:Предприятие (УТ) в файловое хранилище организации должен регулироваться правами доступа на уровне доменных политик, с разрешением только администратору ПК-хоста внешнего ПМ. В права входят: чтение, редактирование, удаление, а также разрешение на копирование в пределах физической памяти ПК-хоста внешнего ПМ. ПК-хост должен представлять собой виртуальную машину и не иметь физического доступа.

Система не ведет работу с личными, а также какими-либо коммерческими данными организации либо её клиентов. Все лица-клиенты в выгружаемом файле (структура в приложении 1) представлены в виде обезличенных идентификаторов системы 1С:Предприятие (УТ).

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении следующих внештатных ситуаций:

- При сбоях в системе электроснабжения аппаратной части, приводящих к перезагрузке ОС, восстановление работоспособности системы должно происходить после перезапуска ОС и запуска исполняемого файла системы

Система не должна иметь следующих возможностей:

- Нарушить стабильную работу сервера 1С:Предприятие (УТ) организации

- Нанести вред данным сервера 1С:Предприятие (УТ) организации
- ## 2.2. Требования к обслуживающему персоналу

Для обслуживания системы во время её эксплуатации определена следующая роль:

- Администратор системы (1 ед.)

Администратор системы должен обладать навыком работ по установке (переустановке) системы, а также знанием структуры системы с её внутренними подсистемами.

Основными обязанностями администратора системы являются:

- Мониторинг работоспособности системы
- Установка (переустановка) системы на ПК-хост при радикальном изменении его аппаратного обеспечения, переустановке ОС или повреждении исполняемых файлов системы

2.3. Требования к видам обеспечения

В требования к видам обеспечения системы входят как требования к информационному, так и аппаратному обеспечению.

2.3.1. Требования к информационному обеспечению

Среди требований к информационному обеспечению присутствуют следующие пункты:

- Внешний ПМ должен поддерживать ОС Windows 7 или 10
- Работа системы должна обеспечиваться средствами языков Python или C#
- ИО между внутренними модулями общей системы должен не покидать пределов локальной сети организации
- Допускается внедрение во внутренние подсистемы средств администратором системы в целях мониторинга работоспособности, если данные средства не передают данные согласно структурам Приложения 1 во внешнюю среду и если они не несут вред общей работоспособности системы

2.3.2. Требования к техническому обеспечению

В состав системы должны входить следующие технические средства:

- Сервер 1С:Предприятие (УТ) организации
- ПК пользователей
- Локальная сеть организации
- Файловое хранилище организации в локальной сети (не менее

1Гб памяти для хранения данных, выгружаемых сервером 1С:Предприятие (УТ))

- ПК-хост внешнего ПМ

Требования к ПК-хосту внешнего ПМ:

- Процессор не ниже Intel Core i5 2,5 ГГц
- Объем оперативной памяти не менее 8 Гб
- Объем дискового пространства не менее 128 Гб
- Сетевой адаптер не менее 100 Мбит

2.4. Требования к документированию

Программный код системы должен сопровождаться комментариями, описывающими принцип работы на русском языке. Система должна иметь документацию для пользователя на русском языке (специалиста поддержки) на этапе эксплуатации и поддержки системы. Итоговый состав документации отражен в таблице 2.

Таблица 2. Состав документации системы

№	Этап работ	Элемент документации
1.	Разработка и апробация алгоритма предобработки и интеллектуального анализа	Комментарии по назначению процедур и функций модуля интеллектуального анализа
2.	Тестирование и запуск системы в работу	Руководство пользователя

3. Состав и содержание работ по созданию системы

Лица, участвующие в разработке системы:

- Заказчик: ООО «Ойл Линк»
- Исполнитель: Миллер Юрий Олегович

В состав работ по разработке системы (таблица 3) входят как первичное изучение предметной области со знакомством ведения автоматизированного процесса и формулировкой технического решения, так и комплексная разработка внутренних систем, механизмов ИО между ними и дальнейшая поддержка системы во время её эксплуатации.

Таблица 3. Перечень этапов работ по разработке системы

№	Содержание работ	Результаты работ	Тип
1.	Изучение предметной области	Описание предметной области и технического решения	Документ
2.	Разработкой механизма выгрузки данных	Доработанная конфигурация на сервере 1С:Предприятие (УТ) (ПО)	ПО
3.	Разработка и апробация алгоритма предобработки и интеллектуального анализа	Внешний ПМ для автоматического подбора, снабженный комментариями программного кода	ПО
4.	Разработка механизма информационного обмена между 1С:Предприятие (УТ) и внешним ПМ	Веб-сервис с API, реализующее ИО между внешним ПМ и конфигурациями клиента 1С:Предприятие (УТ) на компьютерах	ПО

		организации, допускается как часть внешнего ПМ (ПО)	
5.	Тестирование, приемка и запуск системы в работу	Функционирование всех компонентов системы Написание руководства пользователя	ПО Документ

4. Порядок контроля и приемки системы

Виды испытаний системы перед приемкой отражены в таблице 4:

Таблица 4. Перечень испытаний системы перед приемкой

№	Названия испытания	Описание испытания
1.	Тестирование механизма выгрузки данных внешнему ПМ	Проводится при запущенных частях внутренних подсистем: механизм выгрузки данных внешнему ПМ на главном сервере 1С:Предприятие (УТ) организации, алгоритм автоматического подбора в режиме обучения. Система должна успешно произвести обучения алгоритма автоматического подбора, получив выгрузку от главного сервера 1С:Предприятие (УТ) организации.
2.	Тестирование автоматического подбора и заполнения номенклатуры	Проводится при запущенной системе, включая внешний ПМ на ПК-хосте, а также запущенные клиенты доработанной конфигурации 1С:Предприятие (УТ) на ПК пользователей. Система должна обеспечить одновременное выполнение автоматического подбора и заполнения с задержкой не более 10 секунд при отправке данных внешнему ПМ с не менее, чем 5 клиентов доработанной конфигурации.

Приемка системы проводится при участии исполнителя и представителей заказчика в лице генерального директора организации и менеджера отдела продаж в главном офисе организации после прохождения испытаний.

5. Источники разработки

В процессе разработки данного технического задания использовались следующие стандарты:

- ГОСТ 34.602-89 Техническое задание на создание автоматизированной системы

Приложение 1

В данном приложении приводится структура входных данных для функционирования алгоритма автоматического подбора, а также структура данных, являющихся результатом его работы.

1.1 Структура данных при выгрузке механизмом для внешнего ПМ

- ID Клиента (текст, обозначающий клиента, чье содержание заказа сопоставляется с номенклатурой)
- Содержание заказа (текстовые данные на естественном языке, которым оригинально вручную специалистом сопоставляется номенклатура)
- ID номенклатуры (текст, обозначающий сопоставляемую номенклатуру)

1.2 Структура данных, поступающих от пользователей 1С:Предприятие (УТ) алгоритму автоматического подбора внешнего ПМ:

- ID Клиента (текст, обозначающий клиента, чье содержание заказа сопоставляется с номенклатурой)
- Содержание заказа – 1 и более элементов в зависимости от числа позиций (текстовые данные на естественном языке, которым оригинально вручную специалистом сопоставляется номенклатура)

1.3 Структура данных, поступающих от алгоритма автоматического подбора внешнего ПМ пользователям 1С:

- ID номенклатуры – 1 и более элементов в зависимости от числа позиций (текст, обозначающий сопоставляемую номенклатуру)

Приложение 2

В данном приложении приводится пример ассоциации данных при ведении подбора номенклатуры.

ID Клиента	Содержание заказа	Пометка об утверждении	ID номенклатуры (сопоставленной)
<i>УТ-00000922</i>	<i>Счетчик модуль</i>	<i>1</i>	<i>УТ-00007079</i>
<i>УТ-00000822</i>	<i>Соединение 90 ZQ203</i>	<i>1</i>	<i>УТ-00004389</i>

УТ-00007079 - Счетный модуль для М340 2-х канальный, ВМХ ЕНС0200, Schneider Electric.

УТ-00004389 - Штуцер угловой, 90° М22х1,5 R1/4"

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«ПЕРМСКИЙ ГОСУДАРСТВЕННЫЙ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»

**МАТЕМАТИКА И МЕЖДИСЦИПЛИНАРНЫЕ
ИССЛЕДОВАНИЯ – 2018**

Материалы
Всероссийской научно-практической конференции
молодых ученых с международным участием

(г. Пермь, 14–19 мая 2018 г.)



Пермь 2018

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОГО АССИСТЕНТА СПЕЦИАЛИСТА ОТДЕЛА ПРОДАЖ

Миллер Юрий Олегович, Бидуля Юлия Владимировна

Тюменский Государственный Университет, 625003, Россия, г. Тюмень, ул. Володарского, 6,
ym7293@mail.ru

Статья содержит описание подхода к решению проблемы подбора номенклатуры оборудования с использованием ERP-системы. В качестве решения предлагается использовать интеллектуальный алгоритм с применением метода машинного обучения. Предлагается программно-техническое решение для решения проблемы, приведена архитектура и схема взаимодействия модулей информационной системы. Приводится алгоритм машинного обучения, участвующий в функционировании и обучении подсистемы. Указываются результаты работы с демонстрацией применительно к исходному интерфейсу программного обеспечения для специалистов отдела продаж.

Ключевые слова: интеллектуальная система, обработка естественного языка, машинное обучение, парсинг, стемминг, байесовский классификатор, веб-сервис

Введение. На сегодняшний день темпы развития бизнеса и торговых отношений диктуют новые формы ведения работы, при которых интеллектуальные ассистенты специалистов становятся неотъемлемым атрибутом операционной деятельности любой организации. Интеллектуальные ассистенты представляют собой интегрированные в среду работы специалистов интеллектуальные системы (ИС, англ. intelligent system) — это технические или программные системы, способные справляться с задачами, традиционно являющимися творческими, а решение подобных задач считается сугубо человеческой прерогативой. К подобным задачам относятся, например, такие, как задачи распознавания и классификации.

Рассмотрим ситуацию: организация занимается поставкой оборудования и комплектующих на промышленные предприятия. Одним из этапов формирования коммерческого предложения (КП) для контрагента является подбор номенклатуры оборудования в соответствии с заказом клиента, осуществляемый на основании данных ERP-системы. Обычно специалист-эксперт выполняет выбор из каталога существующей номенклатуры, основываясь на собственных знаниях и опыте, обращаясь к внешним источникам, осуществляя непосредственный поиск по совпадению в справочнике номенклатуры для каждой позиции содержания заказа, тем самым затрачивая много времени на процедуру подбора.

Самым простым решением является автоматизация поиска номенклатуры по названию среди данных, хранящихся в ERP-системе. Однако гораздо более эффективная реализация возможна при условии, что будет учитываться предыдущий опыт подбора номенклатуры для

конкретных клиентов. Кроме того, необходимо принимать во внимание возможность несоответствия формулировок содержания заказа известным номенклатурам.

Было принято решение использовать подход, при котором система обучается на текстах ранее сделанных заказов с применением методов математической лингвистики и алгоритмов классификации. В качестве класса принята конкретная номенклатура, в то время как текст заказа является элементом класса. В качестве метода классификации был выбран наивный байесовский классификатор [1].

Алгоритм автоматического подбора номенклатуры включает следующие шаги:

1. Предобработка данных:

- выгрузка содержимого заказов из файлового хранилища (с периодичностью один раз в сутки).

- удаление символов, кроме букв и цифр;

- разбиение на слова-подстроки по пробелам;

- удаление «стоп»-слов (была создана выборка из 250 «стоп»-слов; в них, помимо прочих редко встречающихся в содержаниях заказов слов, входят такие части речи русского языка, как, союзы, частицы, предлоги).

- стеммизация (отсечение окончаний слов для приведения их к словарной форме) с использованием библиотеки PyStemmer.

- представление текста заказа в виде векторной модели, основанной на булевых весах термов [2]. Нуль или единица означают, встречается ли слово заранее подготовленного словаря в содержании заказа. Размерность каждого вектора равна 300. Векторы заказов, отнесенные к известным номенклатурам, формируют обучающую выборку. Кроме того, в качестве признака классификации используется идентификатор клиента-заказчика.

2. Режим обучения

В пространстве векторов обучающей выборки производится вычисление вероятностей для каждого термина с использованием пакета Scikit Learn [3]. Вероятности сохраняются в базе данных внешнего программного модуля.

3. Режим классификации

На вход алгоритма поступает содержание заказа. Оно проходит предобработку, аналогично шагу 1, затем проходит через алгоритм классификации с использованием пакета Scikit Learn. В результате система выдает наиболее вероятный идентификатор наиболее вероятной номенклатуры.

Подбор осуществляется в процессе редактирования документа КП в интерфейсе ERP-системы. В рамках системы клиент и номенклатура характеризуются уникальными идентификаторами. Таким образом, программно-техническое решение должно быть реализовано в рамках ERP-системы с возможностью обмена данными с внешним программным модулем. В качестве ERP-системы используется конфигурация «1С:Предприятие».

На рисунке 1 представлена схема взаимодействия модулей системы. Сервер 1С:Предприятие и рабочие станции специалистов взаимодействуют с внешним программным модулем, включающим веб-сервис с механизмом предобработки текстовых данных и алгоритмом интеллектуального подбора.

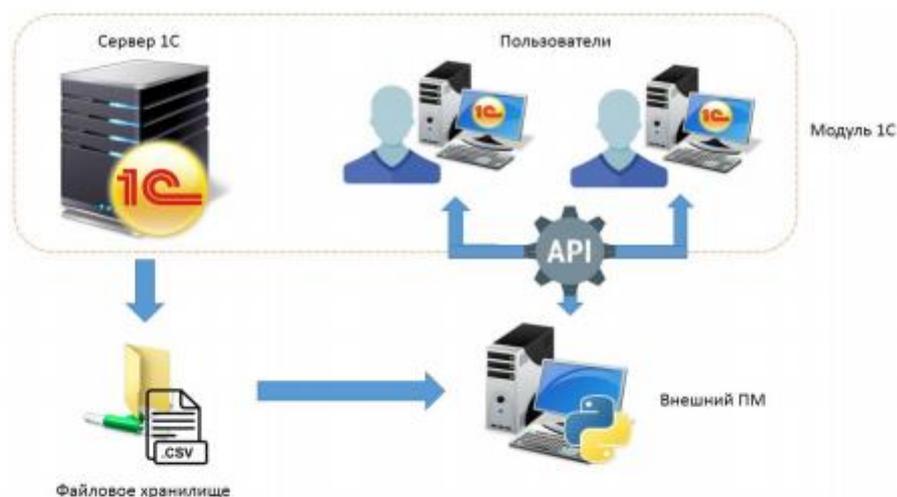


Рис. 1. Схема модулей подсистемы

В конфигурацию сервера 1С:Предприятие были внесены изменения, заключающиеся в добавлении операции выгрузки (регламентного задания) товарных позиций коммерческих предложений, в которых содержанию заказа уже подобрана конкретная номенклатура. Файл-выгрузки в формате «csv» выгружается на файловое хранилище организации и содержит такие данные, как идентификатор клиента, содержание заказа и соответствующий идентификатор номенклатуры.

Внешний ПМ включает в себе механизм загрузки данных из файлового хранилища, веб-сервиса с API для обмена с конфигурацией клиента 1С:Предприятие, а также реализованным интеллектуальным алгоритмом подбора. Внешний ПМ должен возвращать клиентской конфигурации 1С результат работы алгоритма подбора номенклатуры исходя полученного клиента и содержания заказа.

Реализованный алгоритм подбора способен подбирать номенклатуру (рисунок 2), даже если ID (идентификатор) клиента не входил в выборку и окончания слов содержания отличаются. При подаче идентичных по ассоциированной номенклатуре содержаний и разных ID клиента, алгоритм будет подбирать номенклатуру с учетом клиента.

Клиент: ОАО "Компания" | Статус: Действует

Организация: ООО | Действует до: . . .

Операция: Реализация | Валюта: руб. | Текущее состояние: руб.

Содержание: Товары (1) | Дополнительно

Вариант 1 | + Добавить вариант

Добавить | Разделитель | Подобрать через И.Т.А. | Заполнить цены

№	Содержание	Комментарий	Артикул	Номенклатура
1	Счетчик "модуля"			Счетный м...
2	Соединение ZQ203			Штуцер уг...

Рис. 2. Результат работы интеллектуального алгоритма в интерфейсе 1С:Предприятие

По предварительным оценкам, в результате применения интеллектуального ассистента возможно высвобождение до 40 человеко-часов в месяц за счёт автоматизации подбора номенклатуры при составлении коммерческих предложений специалистами отделов продаж.

Библиографический список

1. Machine Learning Mastery - How To Implement Naive Bayes From Scratch in Python [Электронный ресурс] URL: <https://machinelearningmastery.com/naive-bayes-classifier-scratch-python/> (дата обращения: 02.02.2018).
2. ML Wiki - Vector Space Models [Электронный ресурс] URL: http://mlwiki.org/index.php/Vector_Space_Models (дата обращения: 06.02.2018).
3. Scikit learn - Naive Bayes [Электронный ресурс] URL: http://scikit-learn.org/stable/modules/naive_bayes.html (дата обращения: 10.02.2018).

THE DEVELOPMENT OF INTELLECTUAL ASSISTANT FOR SALES DEPARTMENT MANAGER

Miller Yuriy O., Bidulya Yuliya V.

University of Tyumen, 6 Volodarskogo Street, Tyumen, Russia, 625003, ym7293@mail.ru

This paper contains description of the problem of manual stock list association, carrying by managers using ERP systems. The first part of the work presents the relevance of machine learning for purposes of the manual association problem. The second part of this work gives the description of suggesting software and hardware solution. This paper also shows architecture of the above-mentioned solution. The third part demonstrates the main steps during the natural language processing for learning and classification which was implemented in the external program module of the solution. This part shortly describes machine learning algorithm which associates order content position with item from the stock list as a classification problem. The conclusion of this article shows the results of work, including screenshot of the program interface after successful association.

Keywords: intelligent system, natural language processing, machine learning, parsing, stemming bayesian classifier, web-service

```

def work(input):
    input=syn replacer(input)
    client=input.split('|')[0]
    input_array=[set(stemmer.stemWord(k.lower())) for k in
string_replacer(x).split()) for x in input.split('|')]
    inp = input.split('|')
    result_rate=[0]*(len(input_array)-1)
    result_text=['']*(len(input_array)-1)
    result_bool=[False]*(len(input_array)-1)
    #Исторический фактор
    for line in Ord_file:
        if line.count('\t') == 6 and '\t\t' not in line:
            arr = line.split('\t')
            for i in range(1, len(inp)):
                if (" ".join(str(arr[1].lower()).split()) == "
".join(str(inp[i]).lower().split())):
                    rate = 0.0 + (str(arr[0].lower()) == client.lower()) +
(str(arr[2].lower()) == '1') / 2 + compareSizes("
".join(str(arr[1].lower()).split())," ".join(str(inp[i]).lower().split()))
                    if rate > result_rate[i - 1]:
                        result_rate[i - 1] = rate
                        result_text[i - 1] = arr[4]
                        result_bool[i - 1] = True
    result_rate = [0] * (len(input_array) - 1)
    #Лингвистический подбор
    if not checkComplete(result_bool):
        for line in Nom_file:
            if line.count('\t')==3 and '\t\t' not in line:
                id=line.split('\t')[0]
                item=string_replacer(line.split('\t')[1])
                items=list([set(stemmer.stemWord(x.lower())) for x in
item.split()])[0])
                for i in range(1, len(input_array)):
                    if not result_bool[i-1]:#если не подобран ранее, то
подбираем
                        count=0
                        count_of_words=0
                        for word in input_array[i]:
                            count += items.count(word.lower())
                            count_of_words += items.count(word.lower())
                        if str(line).count(str(word.lower())) < 1:
                            count -= 1
                        count+= compareSizes("
".join(str(arr[1].lower()).split())," ".join(str(inp[i]).lower().split()))
                        if count > result_rate[i - 1] and count_of_words>0:
                            result_rate[i - 1] = count
                            result_text[i - 1] = id
    print(result_text)
    return '|'.join(result_text)

```

```

def work_ext(input):
    input = syn_replacer(input)
    client = input.split('|')[0]
    input_array=[set(stemmer.stemWord(k.lower()) for k in
string_replacer(x).split()) for x in input.split('|')]
    inp = input.split('|')
    result_rate=[-99]
    result_text=''
    #Лингвистический подбор
    for line in Nom_file:
        if line.count('\t')==3 and '\t\t' not in line:
            id=line.split('\t')[0]
            item=string_replacer(line.split('\t')[1])
            items=list([set(stemmer.stemWord(x.lower()) for x in
item.split())][0])
            count=0
            count_of_words=0
            for word in input_array[1]:
                count+=items.count(word.lower())
                count_of_words += items.count(word.lower())
                if str(line).count(str(word.lower()))<1:
                    count-=1
            count+=compareSizes(str(item), str(input_array[1]))
            if count > result_rate[0] and count_of_words>0:
                result_rate.insert(0,count)
                result_text.insert(0, id)
            if count == result_rate[0]
                result_rate.insert(len(result_rate)-1, count)
                result_text.insert(len(result_text)-1, id)

            if len(result_rate) >= ext_size: result_rate.pop()
            if len(result_text) >= ext_size: result_text.pop()
        if result_text[len(result_text)-1]==' ': result_text.pop()

    result_text=[x for _x, x in
sorted(zip(result_rate,result_text),reverse=True)]
    #Исторический фактор
    result_r = [0]
    result_t = ['']
    for line in Ord_file:
        if line.count('\t') == 6 and '\t\t' not in line:
            arr = line.split('\t')
            if (" ".join(str(arr[1].lower()).split()) == "
".join(str(inp[1]).lower().split())):
                rate = 0.0 + (str(arr[0].lower()) == client.lower()) +
(str(arr[2].lower()) == '1') / 2 + compareSizes("
".join(str(arr[1].lower()).split())," ".join(str(inp[1]).lower().split()))
                if rate > result_r[0]:
                    result_r[0] = rate
                    result_t[0] = arr[4]
            if(result_r[0]>0 and result_t[0] != ''):
                result_text.insert(0, result_t[0])
        print(result_r)

    print(result_rate)
    print(result_text)
    return '|'.join(result_text)

```

Приложение 6

Листинг 7. Получение значения фактора измерений для рейтингования

```
def compareSizes(input, order):
    rInput = re.search(r'\d{0-9}*\s*(.|\,)*[0-9]*\s*([\s|*|*])\s*\d{0-9}*\s*(.|\,)*[0-9]*\s*\d', input,
    flags=re.IGNORECASE).group()
    rOrder = re.search(r'\d{0-9}*\s*(.|\,)*[0-9]*\s*([\s|*|*])\s*\d{0-9}*\s*(.|\,)*[0-9]*\s*\d', order,
    flags=re.IGNORECASE).group()
    sizesInput = re.split(r'x|x|\*', rInput)
    sizesOrder = re.split(r'x|x|\*', rOrder)
    sizeFactor = 0
    if (len(sizesInput) == len(sizesOrder)):
        for i in range(len(sizesInput)):
            sizeFactor += abs(float(sizesInput[i].replace(',','.')) -
float(sizesOrder[i].replace(',','.')))
        sizeFactor = 1 / sizeFactor
    return sizeFactor
```