

Алгоритм поиска способов устранения неисправностей в системе обслуживания листовых офсетных печатных машин

Д.А. Денисов,

аспирант кафедры информационных технологий
Московского государственного университета печати имени Ивана Федорова

В данной статье описывается алгоритм поиска способов устранения неисправностей, возникающих при работе производственного оборудования. Если в процессе производства возникают неисправности, устранить которые оператор не может по причине невысокого уровня компетенций, то для решения этой задачи оператором может быть использована экспертная система, позволяющая находить наиболее подходящие способы устранения. Процесс поиска начинается с того, что оператор формирует запрос в свободной форме, указывает признаки неисправности. Перед разработкой алгоритма была сформулирована задача по интеллектуализации поиска. Под интеллектуализацией поиска понимается:

1. Поиск описания неисправности, учитывая однокоренные слова.
2. Поиск описания неисправности, учитывая синонимы, так как запрос может быть сформирован в свободной форме.

Для решения задачи необходимо ввести дополнительные множества:

$S = \{S_i | i = 1, s\}$ – множество слов в запросе оператора листовой машины;

$Y = \{Y_i | i = 1, y\}$ – множество корней слов;

$X = \{X_i | i = 1, x\}$ – множество слов, входящих в описание неисправности в базе знаний;

$K = \{y \mid P(y)\}$ – множество совпавших корней;

$L = \{K_i \mid Z(K_i)\}$ – расширенное множество K , содержащее корни слов, которые являются синонимами к словам, корни которых содержатся во множестве K , где Z и P – характеристические предикаты.

Множество S содержит в себе перечень слов, которые ввел оператор при формировании запроса. Множество корней содержит в себе корни слов, которые наиболее часто употребляются при описании неисправностей. В качестве примера можно привести такие корни слов, как {мат, бум, лист, тиск, мят, форм, ниж}, соответственно от слов {материал, бумага, лист, оттиск, мятый, форма, снижение}.

До выполнения процедуры сравнения из строки запроса программой удаляются все слова, длина которых менее или равна трем символам (предлоги, союзы) для уменьшения времени процесса поиска. Использование данного алгоритма позволит компенсировать некомпетентности оператора за счет сопоставления корней синонимичных слов. Например, оператор описывает неисправность, используя не самые распространенные и принятые к употреблению слова. Система анализирует их, находит однокоренные и производит поиск на основании большего множества слов, нежели чем изначально было указано оператором.

Алгоритм поиска способов устранения неисправности на основании данных о внешних признаках ее проявления осуществляется путем сравнения каждого элемента множества Y с элементом множества S .

Если при последовательном анализе обнаруживается, что множество Y_1 включено во множество S_i , то элемент множества Y удаляется, если же множество Y_1 не включено во множество S_i , то множеству Y_1 присваивается индекс последнего элемента, а индексы других элементов изменяются на их новые порядковые номера.

Работа алгоритма заключается в циклическом выполнении операций поиска включения одного множества в другое. Завершение работы цикла осуществляется, когда произошел перебор всех элементов из множества Y .

В случае, если элемент множества Y входит в любой из элементов множества S , то он добавляется в отдельное множество K , то есть K – множество корней слов, которые были найдены в исходном запросе. На следующем шаге, после анализа всех слов из запроса, необходимо расширить множество корней для более точного и обширного поиска, добавив в него корни синонимичных слов, информация о синонимичности слов которых содержится в специальной таблице соответствия корней слов-синонимов. На данном этапе создается множество $L = \{K_i \mid Z(K_i)\}$, элементами которого также являются множества символов. На следующем этапе необходимо выявить вхождение корней из расширенного множества L во множество X . Условие вхождения корней

из расширенного множества в словах строки с описанием, находящейся в базе знаний, описывается следующим уравнением:

$$L_i \subset X_j \forall l_n \in L_i : l_n \in X_j,$$

где l_n — элемент множества L_j .

Если при последовательном анализе обнаруживается, что множество L_i включено во множество X_j , то элемент множества L_i удаляется, значение параметра l , учитывающего количество совпадений корней со словами увеличивается на единицу, а индексы оставшихся элементов множества L уменьшаются на единицу. Если множество L_i не включено во множество X_j , то множеству L_i присваивается индекс последнего элемента, а индексы других элементов уменьшаются на единицу.

Если оператор ищет способ устранения и в запросе описания неравности указывают слова, корни которых не входят во множество Y , а способ устранения возникшей неисправности в процессе выполнения алгоритма поиска был успешно найден, то слова, корни которых не будут найдены, обрабатываются на основе лингвистического анализа, записываются в базу знаний. Из слов будут выделены корни и подобраны корни слов-синонимов. Таким образом, системой производится самообучение, позволяющее накапливать слова. Накопленные знания позволяют более точно находить способы устранения. Для лингвистического анализа используются два словаря: словарь корней слов и словарь синонимов.

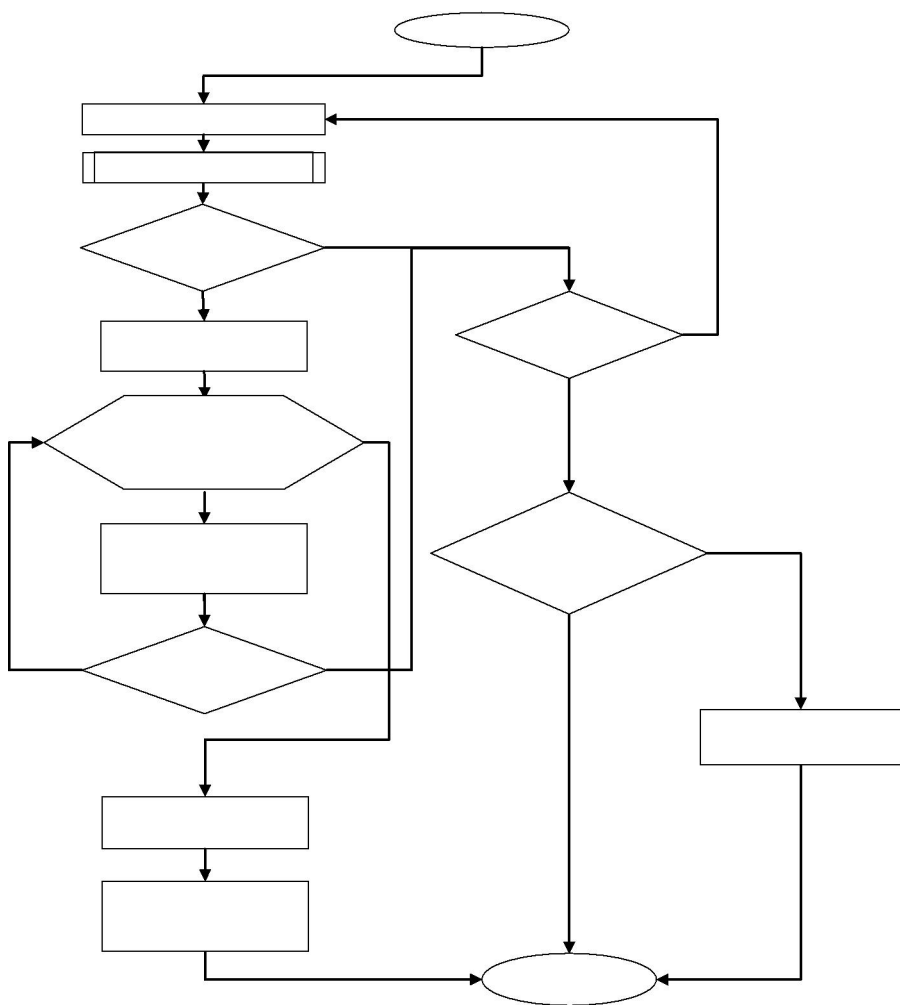
Также ставится задача эффективного использования полученной информации оператором, чтобы максимально ускорить процесс устранения причины неисправности. Для этого необходимо вывести тот способ устранения, который бы был наиболее вероятный. Наиболее вероятным считается тот способ, который наиболее полно совпал со строкой запроса, степень совпадения вычисляется в процентах. Если было выявлено несколько описаний неисправностей с одним процентным показателем, то приоритет будет отдан тому способу устранения, который имеет более высокий рейтинг.

Процент совпадения вычисляется по формуле

$$Q = \frac{I}{J} \times 100\%,$$

где I — количество элементов множества X , которые включают элементы множества L ; J — количество слов в описании неисправности в базе знаний; Q — процент совпадения.

В случае, если способ неисправности найден не был, то оператору предлагается написать запрос в службу поддержки или сформировать запрос иначе, как изображено на схеме алгоритма (рис. 1).



(Продолжение рисунка на стр. 201)

Библиографический список

1. *Попов Д.И.* Программирование на языке высокого уровня Паскаль / Д.И. Попов. — М. : МГУП, 2009.
2. *Финн В.К.* Искусственный интеллект. Методология, применения, философия / В.К. Финн. — Красанд, 2011.
3. *Джарратано Д.* Экспертные системы. Принципы разработки и программирование / Д. Джарратано, Гари Райли. — М. : Вильямс, 2011.
4. *Макконнелл Дж.* Анализ алгоритмов. Активный обучающий подход / Дж. Макконнелл. — М. : Техносфера, 2011.