IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 36, No. 2, April, 2022, pp. 297-304
Journal homepage: http://iieta.org/journals/ria

Procedural Knowledge Mining - A New Method for Extracting Best Practices by Applying Machine Learning on Data Graph

Souaad Hamza-Cherif*, Azzedine Chikh

Laboratory of Research in Informatics, Abou Bakr Bekaid University, Tlemcen 13000, Algeria

Corresponding Author Email: souad.hamzacherif@univ-tlemcen.dz

https://doi.org/10.18280/ria.360214

Received: 21 December 2021
Accepted: 16 March 2022

Keywords:
knowledge mining, good practice, data graph, word embedding, unsupervised clustering, text synthesis

ABSTRACT

In recent years with the increase in sharing tools and sites such as Meta, Twitter, WikiHow..., the web has become a constant and permanent source of scalable knowledge where users share their know-how in the form of procedural knowledge. This procedural knowledge, which consists of a successive set of steps for achieving a specific goal, is called good practice. Extracting and formalizing these good practices is a major asset in the field of artificial intelligence. In this context we present a new method for formalizing good practices extracted from the web, and extracting the best practice for a given request by applying the techniques of artificial learning and text summary on data graphs.

1. INTRODUCTION

Nowadays, the amount of information shared on the web far exceeds our ability to reduce and analyze it without the use of automated analysis techniques, especially with the advent of the social and semantic web and different technologies. Among the shared data, we find more and more the procedural knowledge (PK), known also as good practices, which takes precedence in online communities of practice [1]. These good practices are defined as working methodologies composed of a successive set of steps, which have been proven to achieve the desired objective. The major challenge in this case is to extract this PK in order to automate it and facilitate its reuse. According y, PK mining is the research field which offers automated analysis solutions. It uses a set of techniques and algorithms resulting from artificial intelligence in order to extract implicit and potentially useful knowledge, which can serve as a support to the decision-making process [2].

Several research works have addressed the problem of the PK exploration which has become essential for current advanced applications such as SIRI; or ALEXA: the voice assistants of Apple and Amazon. PK exploration is also used in the field of robotics; for example, Tenorth et al. [3] designed a robot that exploits PK of the web to achieve a crepe. Another example of application of such PK are the current information retrieval systems, which exploit knowledge graphs by Chu et al. [4] to answer the requests of new users who seek more explicit knowledge about the way of doing things rather than basic information (date, time, etc.).

In this article we propose a new method to extract the best practice for a given query in order to automatically assist the user in their research process. Differentiating between a good practice and a best practice is a fairly subjective thing. Indeed, there is no formal definition in the literature of a best practice: if a good practice is certainly a practice that works and has been proven to achieve a certain objective, a best practice would therefore be a good practice sorted, selected and considered as better than the others.

The new method uses various techniques, such as web scrapping; artificial learning; text summary; and graph theory, in order to achieve a series of steps:

(1) extracting good practices from the web;
(2) formalizing them as a knowledge graph;
(3) and then exploring them to find the best practice for a given query, by relying on indicators that measure the notion of importance in a graph and identifying popular peaks.

This body of this article is structured in three sections. Section 2 discusses the various related works. Section 3 presents our proposed method in two parts: (1) how to represent good practices by using data graphs; (2) how to extract best practices by applying artificial learning techniques on data graphs. Section 3 shows the experimentation carried out on real data extracted from WikiHow.

2. RELATED WORKS

Good practices or PK, have become the very essence of modern applications, robotics and current search engines. Exploring this knowledge has become a major challenge today. Several research works have focused on the tasks of extracting and exploring PK. Although the type of data processed differs from one approach to another for instance according to Kiddon et al. [5], these data take the form of a cooking recipe, and procedural manuals [6], and the common point is that it represents PK that brings together a set of successive steps to achieve a given objective. From our point of view we define PK as good practices within communities of practices, representing any know-how, formalized as a set of successive steps to achieve an objective.

Data collection is the first step in any knowledge exploration process. Most of the work in the field, like [5, 7-9], use free data available on sites or collected manually. Yang et al. [6] proposed a hybrid technique where the first part of the extraction of the knowledge, modeled as graph, is done

manually and the second part is done automatically using the sequential labeling method which associates each word of a sentence with a tag. The extraction of relations between nodes is based on a current probabilistic measure of the strength of association between two terms, called point mutual information. In our case it would be laborious to manually extract part of the knowledge. Indeed, the nodes of our graph represent good practices including their stages and the relations between nodes represent temporal order of stages in the text. For example, by reading a recipe we see that you cannot put a cake in the oven before you have mixed the ingredients.

Other works like [4, 10] apply automatic tools, to extract data, such as the Stanford parser tool which browses web pages and extracts dependency trees [11]; and the OpenIE tool: Open Information Extraction systems, which allow to obtain the tuples "subject, predicate, object" from a given text [12]. However, these tools, which support designers in collecting information, quickly find their limit. in terms of functionality. For our part, we propose to feed a web scraping algorithm to extract the right PK from the shared website. The program explores the tree structure of web pages in order to extract the good practices and to represent them as a data graph. We opted for data graphs because their advantages in terms of flexibility; ease for updating; ease for calculation; and possibility to identify the measure of importance, etc. Here again the representation formalism differs from one approach to another: a situation ontology is created by Jung et al. [10] in order to model the dynamic aspect of PK but the inference on this type of ontology seems costly in time and memory space. A PK base is created by Chu et al. [4] to feed web graphs. This base is structured as a taxonomy of five elements: task; participant object; participant agent; place; and time. A meta-model is developed by Park and Motahari Nezhad [7] in order to describe a procedure in the form of a directed graph, where the objective to be achieved is represented by the main node, and the secondary nodes represent the elements: method; task and sub-actor; actor; time; location. This representation seems to be close to ours, except that our model only concerns good practices and stages they include, independently of the temporal constraints. Indeed, our major objective is not only the creation of a knowledge base but the extraction of the best practice based only on the stages it goes through.

Different techniques are used in the exploration task depending on the objective of knowledge discovery. There are many methods that seek to extract or identify entities in the text such as the work in:

- Feng et al. [8] which is 'supporting' artificial learning by reinforcement to extract the names of actions and their arguments;
- Gupta et al. [9] which implements a baseline that uses a naive Slot-Grammar-based classification rule to extract instructions and decision points;
- Jung et al. [10] which proposes a learning method not applied to a syntactic model, and the “CRF” Conditional Random Fields to extract actions in verb form;
- Kiddon et al. [5] where we use an undirected learning algorithm on a segmentation model to extract verbs and arguments from the text;
- Park and Motahari Nezhad [7] which uses end-to-end neural networks based on 2 models: HAE model snap into LSTM neural networks (long short term memory), to model the state of sentences in a text, and then sets up a word attention mechanism that captures the most important semantic information conveyed in a sentence. The MemoryNet model independently computes as input the vectors of phrases and stores or downloads them as needed.

All these methods fall within the domain of natural language processing (NLP) and their purpose is the syntactic analysis of the text in order to identify the actions within good practices. However, the goal that we pursue is different since we extract and model good practices directly by exploring the tree structure of web pages represented as semi-structured data.

Noura et al. [13] use the unsupervised K-means clustering algorithm, based on the Word2Vec similarity measure [14], in order to identify popular topics in clusters. The exploration task we are tracking doesn’t focus on the popularity of a subject but rather on the superiority of the practices. We also use unsupervised learning methods based on word embedding: specifically we use ourselves the measure of semantic similarity Word2Vec but in order to group practices similar to the objective sought by the user.

Chu et al. [4] use a hybrid clustering in 2 phases based on 3 different similarity measures: the Wu-Palmer measure; Word2Vec measurement; and vector similarity, to group synonymous tasks and disambiguate them. We then carry out the grouping of similar steps, but our technique does not require an ascending and descending phase during the clustering of the tasks. It is done in a single phase by the unsupervised algorithm DBSCAN [15], based on Word2 with what is less time consuming. Moreover in [4] the grouping is done during the design of the knowledge base. In our exploration task we do not group the practices similar to each other in the knowledge base, because whatever type of clustering is used, it will generate a loss of information. Therefore, the grouping is done during finding the best practice.

Regneri et al. [16] use the ASM multiple sequence alignments to extract the events in the graph nodes and the Wordnet based similarity measure to merge similar nodes, except that to two or more nodes they choose, among all the steps grouped together, the one that will name the new node in the graph. Once again our method differs since we use techniques of text summary. Indeed, we use the PageRank ranking algorithm by Brin and Page [17] to classify the similar stages and thus choose the one with the highest score, in order to represent the new node in the graph of good practices.

3. PROPOSED METHOD

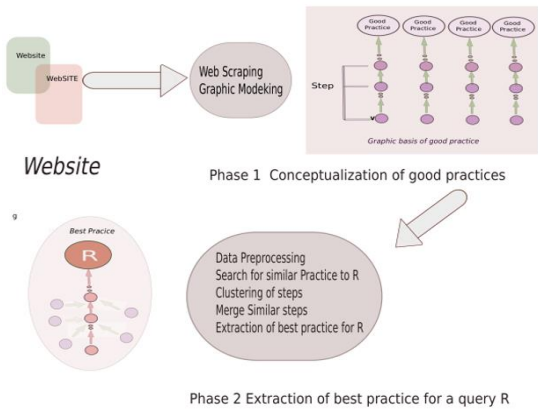

Figure 1. Method overview

In this section we present a new two phase method for conceptualizing good practices in order to extract the best one according to the goal sought by artificial learning on graphs. The procedure is sequence of two phases, as illustrated in Figure 1. In phase 1 we collect data, which we modeled as a directed graph in order to build a base of good practices. In phase 2 we extract the best practice for a given user query from the previous set of good practices obtained in phase 1.

A detailed description of each phase is given in the following subsections.

3.1 Conceptualizing good practices

This phase includes two major steps: data collection and modeling of good practices.

3.1.1 Data collection

We apply an automated technique for extracting content from Web, called web scraping [18]. It focuses on transforming semi-structured data on the web, usually in HTML format, into structured (CSV format i.e. Coma Separated Values) data that can be stored and analyzed in a local database. It is necessary to launch an http request in order to parse the web content. In this way, HTML or XML documents are analyzed as a tree structure.

3.1.2 Modeling good practices

To model good practices we use oriented data graphs. The advantages of such a representation are undeniable. It allows to represent any object as node and any property as arc. Moreover, the graph is very flexible; one can increase it without attenuating it. A directed graph of good practice G is defined by a pair of sets V and E, noted G = (V, E), such that: V is the set nodes representing good practice and its sequential steps, and E is the set of arcs that connect nodes. The arcs in the graph are directed in such a way that they connect the steps of a practice to the goal to achieve by this practice. The direction of the graph is represented by incoming arcs starting from the first step (which represents a parent node without a predecessor) and going through all the intermediate steps until the last step, this one is then connected by an arc entering to the practice which is represented by a node without successor.

Figure 2 presents an example of modeling a good practice "Make a cake" This practice includes 5 successive steps: “prepare the ingredients; “preheat the oven”; “Mix the ingredients”; “bake in the oven”; “Unmold the cake”.

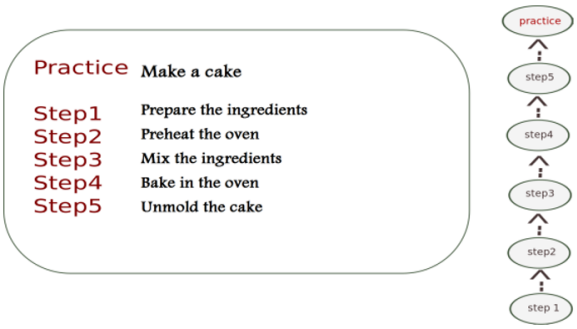

Figure 2. Example of good practice modeling

3.2 Extracting the best practice

This phase consists of extracting the best practice among those existing in the knowledge base for a given query. Considering the huge and increasing amount of PK shared in the web community, it is difficult to find the best among all existing ones. For example, if we launch a query to find a recipe for a cake, we would be faced with the dilemma of choosing among all of the returned results. Therefore identifying the best PK to bake a cake becomes laborious. One can certainly refer to online rating systems but again this requires analysis and evaluation of all ratings and comments to form an opinion. Differentiating between a good and a best practice therefore remains subjective. In our method, we consider the best practice as the practice having the most used steps by all the good practices with the same request. In other terms it is the practice passing by the most important steps represented as nodes. If a step in a practice is frequently used by other practices that seek to achieve the same request, then this step is considered as important in satisfying the request. Therefore, the importance of those steps can be quantified by the measures of centrality in the graphs which can be interpreted as a high degree of incoming centrality. This phase includes three stages:

(1) The first stage aims at searching for similar practices, regarding a user request "r". We first look for all the practices having the same goalas "r". Hence we use an intuitive mathematical method based on “word embedding” to group together all practices semantically close to "r".

(2) The second stage aims at merging similar nodes which represent identical steps. We use an unsupervised DBSCAN clustering algorithm [15] to group the semantically close nodes. Then we use a text synthesis technique to merge the resulting similar nodes.

(3) The last stage aims at identifying the best practice that satisfies the user's request “r” from the new graph G. In this case we compare the importance of each path in the graph, which is visited to satisfy the request “r”. This importance measure is based on the paths crossing the nodes with the greatest number of incoming arcs to reach “r”.

Figure 3 shows the phase of extracting the best practice for a query. We explain below in more details each stage of this phase.

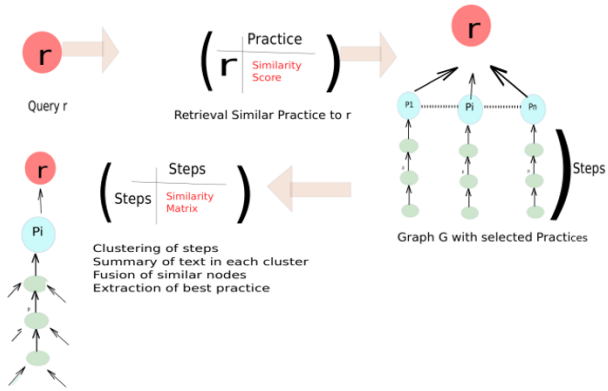

Figure 3. Phase of extracting the best practice

3.2.1 Searching for similar practices

Classical information retrieval generally relies on various supervised and unsupervised classification methods or similarity measures in order to return documents semantically close to the user's request. In our case, the proposed method is inspired from the WMD "word mover's distance" method [19],

which exploits advanced integration techniques such as word embedding, to calculate the semantic distance between text documents So given a user request, we consider each practice title in parallel with the words making up the request. After a series of pre-processing and cleaning (blank word deletion, lemmatization, tokenization) we calculate the semantic distance of the word vectors obtained using the Word2Vec word extension model [14]. The former is a two-layered neural network trained to predict the vector representation of words in context. More simply Word2Vec takes as input a text corpus and outputs a set of vectors for the words of this corpus, in order to group vectors of similar words in vector space.

This machine learning requires a training base. We will then use the Google learning model [20] available on the web, made up of 300 vector dimensions for 3 million words and sentences. We will select the practices having vector representations closest to the query. Hence we define a function F (x, y) between [0,1] which returns the rate of similarity between two sentences x and y on the basis of the proximity between their vector representation using the Word2vec template [14]. We define a similarity threshold s = 0.7, beyond which two words are considered to be semantically close. We thus calculate the similarity score between the query and the existing good practices. Only the practices having a similarity score greater than the threshold will be selected. At the end of this stage we model a new graph G where the main node represents the request of the user r to which will be connected by incoming arcs all the selected good practices and the steps they encompass.

3.2.2 Merging identical nodes

After having selected the practices similar to the request sought by the user, we end up with a new graph G that may contain similar steps That is to say steps that are syntactically different but semantically identical, it would therefore be judicious to merge them. To do so we first group similar steps into separate clusters and then determine for each group of steps which sentence will summarize all of the steps to be merged into an end node using a text summary technique:

Grouping similar steps. To group similar nodes together, the unsupervised DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm is used, which has the advantage of being efficient in terms of computation time without needing to predefine the number of clusters. This algorithm is used to cluster high density data points and does not take into account outliers in low density regions [15]. It has two main parameters: "eps" which determines the threshold above which 2 points are considered as neighbors and "min-point" is the minimum number of points to form a dense region.

The DBSCAN algorithm takes as input the square matrix D of distances or dissimilarity between the steps of the graph G, which is equal to the complement of the square matrix of similarity S (D = 1-S). The similarity matrix S is calculated by using the same function defined in the previous paragraph F (x, y), (included between [0,1]), which returns the similarity score between two sentences x and y S(x,y) by applying the Word2Vec model [14]. Also in this case we only consider the similarity scores above the threshold s (s = 0.7):

(1) If F (x, y)> = s: S (x, y) = S (y, x) = F (x, y).

(2) Otherwise S (x, y) = S (y, x) = 0.

Text summary. It is an important task of machine learning and NLP language processing It refers to the technique of shortening large texts with the aim of creating a coherent and fluid summary containing the main points highlighted in the document. There are two main types of NLP text synthesis: extraction and abstraction. Extraction methods select a subset of existing words, phrases, or sentences in the original text to form a summary. In contrast, abstraction methods first build an internal semantic representation, and then use natural language generation techniques to create a summary. Most text synthesis systems are based on some form of extractive synthesis [21, 22].

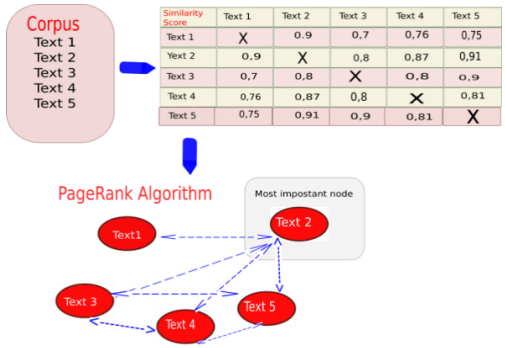

Figure 4. Example of text summarization technique

In our method, we use text synthesis to merge the nodes representing the similar steps; so we search among each set of grouped steps the lexical representation closest to the others. For this, we use a similarity extraction method that identifies the most important points of a set of sentences. For each group of steps obtained during the clustering process, we first build a similarity matrix between the sentences representing each node (step) of the cluster, based on the Word2Vec learning model [14] From this matrix we generate a data graph where the nodes represent the sentences of the corpus and the arcs the links of similarity between these sentences. We then apply the PageRank classification algorithm [17] to classify the most important sentences. Recall that this algorithm identifies a node as being important if it is pointed to by other important nodes. From there we can identify the most important sentence in each set of steps that is the one that will have the most incoming arcs and thus we get the lexical summary of the steps grouped together. Figure 4 schematizes an example of a text synthesis process for similar nodes.

3.2.3 Extracting the best practice

In this last stage, we use data graph theories to extract the best practice for a given query r. More precisely, we use indicators that measure the notion of importance in a graph by identifying popular peaks. This notion of importance represents a major stake in several fields and in particular in the analysis of social networks, where we seek to identify online communities and influential people thanks to various measures of centrality of graphs [23]. In search of information the ranking of the results is also influenced by the degree of importance of a document. The best example remains the famous algorithm PageRank of Google [17] which says that a page is important if it is tagged by other important page.

In our method we also exploit this notion of importance of vertices in graphs to evaluate good practices. The best practice for a given query will be the one with the path grouping the most important vertices. In order to quantify the importance of each path leading to the desired objective, we base ourselves on the degree centrality measure, also called the prestige measure [24]. This measure is the simplest form of the notion

of centrality. It is based on the idea that the importance of a node depends on the number of its neighbors and therefore on the number of its incident links. In graph theory, this number is called the entry or exit degree of a node, which is measured either by the number of its incoming arcs (the incoming degree of centrality) or by the number of its outgoing arcs (the outgoing degree of centrality).

In our method we will compare the importance value of each path in order to identify the best practice. To quantify this importance we calculate the ratio between the sum of the incoming arcs of the nodes that each path traverses with the total number of nodes traversed. More formally we define the importance of each path by the Eq. (1):

$$Importance_{Value}(Pj) = \frac{\sum_{i=1}^{n} Id(i)}{n} \quad (1)$$

With:

Pj: the path j that leads to the goal sought by the user's request,

n: the total number of nodes traversed by the path Cj to reach the desired goal,

Id (i): the incoming degree of node i which results in the number of its incoming arcs.

4. EXPERIMENTATION AND RESULTS

Here, we present the experiment that we carried out on real data from the Web. This section is divided into two subsections.

In the first we present the dataset extracted automatically from the WikiHow sharing site [15], which we formalized in the form of a data graph.

In the second we test the feasibility of our method based on node clustering and text summary technique son a concrete example of our dataset.

4.1 Conceptualizing good practices in WikiHow

In order to test our method, we automatically extracted a dataset from the WikiHow sharing site. This collaborative platform of the net, created by Jack Herrick in 2005 [25] for automatically exchanging data and information around the world, provides a perfect framework for exploiting PK. It's a collection of information called also “How-to” that allows finding many alternative solutions to a specific problem. WikiHow platform is composed of a set of articles grouped by category. Each article contains a set of practices described by a set of sequential steps. Figure 5 shows the structure of the WikiHow site.

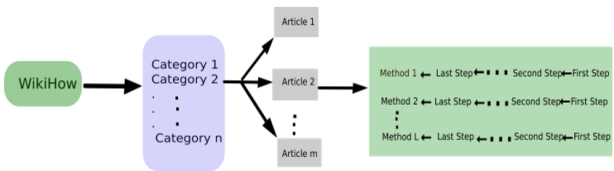

Figure 5. WikiHow structure

Our web scrapping algorithm is based on the Beautiful Soup and Lxml [26] python libraries which are placed as html and xml parsers in order to analyze the tree structure of a web page and search for the necessary data. The scraping algorithm searches for good practices in articles by category. For each category we extract all the items it contains, and for each article we extract all the practices and steps it describes. For our experimentation we are interested at the categories relating to the field of health and its derivatives. The result of the extraction is shown in Table 1.

Table 1. Result of WikiHow extraction

Number of categories	407
Number of articles	9659
Number of practices	27266
Number of steps	348634

The set of extracted practices and steps necessary to achieve each one of them, were modeled as oriented data graphs, where each good practice represents a parent node preceded by a set of successive nodes corresponding to the steps necessary to complete the practice.

4.2 Extraction of best practices

In this section we present the results obtained from a series of experiments carried out at each level of the best practice extraction phase (searching for similar practices, grouping similar nodes, and identifying the best practice):

4.2.1 Evaluation of the research system for similar practices

We evaluated the performance of our information retrieval system based on its ability to return relevant documents for a given query. In classical information retrieval the two main factors for evaluating IRS (information retrieval systems) are: the recall which signifies the capacity of a system to select all the relevant documents from the collection; and the precision which is the capacity of a system to select only relevant documents. However this requires having a pre-established model to perform the comparison, which is not the case for us. Therefore our assessment is based on a comparison between methods in the field, for this to do so; we have implemented search algorithms for similar practices using the two variants of similarity measurement, namely:

(1) Word Moving Distance WMD [19], which exploits advanced integration techniques such as word embedding to calculate the semantic distance between text documents;

(2) Term Frequency - Inverse Document Frequency TFIDF [27] which returns the similarity between two lexical entities in calculating the frequency number of a word in a document.

We launched different queries in the different implemented systems, and we noticed that the practices returned by our method were more relevant for in comparison with those returned by WMD and TF-IDF. We find in Table 2 an example of the first 10 results returned for a request R1 “how to treat flu” for each implemented method. We notice that, among the best results returned by WMD, practices, such as “Meditate for Health”; “Be a Health Nut”; or “Do a Butterfly Stretch”, are not correlated with the searched request. The most similar practice to the query “treating the flu” is only ranked in the 9th position, which is quite aberrant as a result. On the other hand in TF-IDF, which tends to return better results than WMD, we find in first position of the ranking the same practice returned by our method. However the following practices in the rankings look less relevant. For example, we find in 2nd and 3rd positions practices such as "Preventing the flu"; or even "identifying the flu", which are less relevant for the request. In our method we find in 2nd and 3rd positions practices like

"Treating the Flu with Supplements" and "Treating a Cold or Flu at Home", which are more relevant to the launched query.

Figure 6 gives the similarity scores of the ten best practices returned by the three methods regarding the query R1, and for two other queries: R2, "how to be healthy and R3 "how to manage stress". We remark that our method surpasses the two others, to capture and return the relevant results with higher similarity rates.

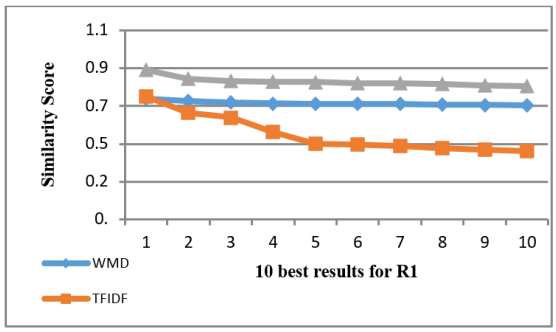

Figure 6. Curves representing the ten best similarity scores obtained by the three methods for the request R1

For quantitatively evaluating the advantages of our method, in addition to the similarity scores, we compared the average execution time for a query R1 for the three methods: WMD, TF-IDF and ours.

Table 2. The 10 best results obtained by using the three methods: WMD; TF-IDF; and our method for query R1

Method	10 best results
WMD	Meditate for Health Be a Health Nut Do a Butterfly Stretch Dealing with the Aftereffects Inserting the Catheter Treating the health problem Treating the Cut Start a Healthy Diet Treating the Flu Create the Chart
TFIDF	Treating the Flu Understanding the Flu Preventing the Flu Identifying the Flu Diagnosing the Flu Treating the Flu with Supplements Recognizing Flu Symptoms Getting the Flu Vaccine Fighting the Flu with Food Preventing the Common Cold and Flu
Proposed method	Treating the Flu Treating the Flu with Supplements Treating a Cold or Flu at Home Diagnosing the Flu Getting the Flu Vaccine Understanding the Flu Identifying the Flu Preventing the Flu Treating Flu Symptoms with Medicinal Remedies Deciding When to Get the Flu Vaccine

Evaluation results are presented in Table 3. We clearly see that the execution time of our proposed method is much shorter than the WMD time and very close to TF-ID time. These results are very encouraging considering the performance of our method.

Table 3. Execution times of the three methods: WMD; TF-IDF; and ours for query R1

Method	Execution time
WMD	98,96766067 seconds
TFIDF	1,466084003 seconds
Proposed method	1,899108648 seconds

4.2.2 Clustering evaluation

We seek to measure the quality of our partitioning and to study the impact of the similarity metric on our classification algorithm. Therefore we use the silhouette coefficient [28], which calculates the quality of a partition of a dataset in the domain of automatic classification. The silhouette score varies within the interval [-1.1]:

- 1: Signifies that the groups are well separated from each other and clearly distinguished;
- 0: means that the clusters are indifferent, or we can say that the distance between the clusters is not significant;
- and -1: means that the clusters are allocated in the wrong direction.

4.2.3 Evaluation of the best practice

We have parameterized DBSCAN with: the precomputed metric, whose input is the square matrix of distance or dissimilarity obtained by the Word2Vec model, and the cosine metric which calculates the distance between two vectorized points thanks to the TF-IDF measure.

We then compared DBSCAN to another Kmeans unsupervised classification algorithm [29]. This algorithm requires choosing in advance the number of clusters K to do the data partitioning.

The results of the experiment obtained for three sample requests: R1 "How to treat the flu"; R2 "how to be healthy"; and R3 "How to manage stress" are shown in Table 4. We clearly see that our method outperforms the others in terms of clustering quality. Indeed, we arrive at scores of 0.649 for the second request which is quite good. It is true that in the first and third requests, our score is around 0,5but it remains much higher than the others. Indeed, since the quality of the clustering in the case of textual data which is not quantitative but rather qualitative.

Table 4. Silhouette score results

Clustering	SS(R1)	SS(R2)	SS(R3)
DBSCAN (metric precomputed)	0.554	0.649	0.522
DBSCAN (cosine metric)	0.243	0.351	0.241
Kmeans	0.149	0.565	0.329

Another finding is that the Kmeans algorithm in the two last queries surpasses that of DBSCAN using the Cosine metric. Thus we can confirm that the metric strongly used impacts the quality of DBSCAN partitioning. In this experiment, DBSCAN with the precomputed metric outperformed Kmeans.

4.2.4 Best practice assessment

In this subsection we try to check the consistency of the results found in terms of best practice. As we had already specified, there is no precise way to determine the superiority of a PK on others. Also we don't have any predefined

evaluation model, then checking the consistency of the results is quite tricky. In this evaluation scenario we consider the R1 query "how to treat the flu". We compare the results obtained using our method with the results obtained using the WikiHow platform. We have thus selected the article (we consider all the practices described in the article) returned at the top of the list which serves as the best results for our comparison, the results obtained are presented in Table 5.

From these results, we note first that the best practice extracted by our method "Treating the flu" is more relevant than the best practices returned by the WikiHow site such as "Identifying the flu", or "Preventing the flu", that don't represent relevant results for the request searched.

Then, we note that the article returned at the top of the list by WikiHow contains four different practices from each other; each one addresses an aspect of the domain of the request while our method focuses on extracting a single practice which is considered the best.

Table 5. Best practice results returned by WikiHow and our method for request R1

Method	Results
Proposed method	**Best practice**: Treating the Flu **Steps:** Rest - Stay hydrated and avoids cigarettes and alcohol - Fight a low fever with OTC medication -Take cold medication for other symptom - Identify a dangerous fever based on age - Watch for warning signs - See a doctor early if you are at risk of complications.
Wikihow	**Best practice 1**: Identifying the Flu **Steps:** Recognize the symptoms of the flu - Distinguish between the flu and a cold. - Distinguish between the flu and a stomach bug - Know when to seek emergency medical treatment. **Best practice 2**: Treating Flu Symptoms with Natural Remedies **Steps:** Get some rest - Stay hydrated - Take a vitamin C supplement - Clear mucus from your nose often - Use a heating pad - Relieve fever symptoms with a cool cloth - Gargle with saltwater - Try an herbal remedy to relieve your symptoms - Try a eucalyptus steam treatment. **Best practice 3**: Taking Medication to Treat Your Symptoms **Steps:** Buy over-the-counter medicine to treat symptoms - Give children the correct dosage - Take prescription medication as directed - Understand that antibiotics will not treat the flu. **Best practice 4**: Preventing the Flu **Steps:** Get vaccinated before flu season -Talk to your doctor before getting the vaccine if you have certain conditions - Choose between the flu shot and the nasal spray vaccine - Practice good hygiene -Keep your body in good general health - Take the flu seriously.

We also notice that the best practices returned by WikiHow include some steps which are redundant, for example in the best practice returned in the article "Identifying the Flu" (if we refer to the order of ranking in the WikiHow article) we find stages like "Recognize the symptoms of the flu" and "Distinguish between the flu and a cold" which are similar. On the other hand as in our method we carry out the grouping of similar steps, this allows to avoid the redundancy of the steps to follow in the extracted best practice for a given request.

We have run our experimentation on a series of queries and each time our method outperforms the results returned by WikiHow in terms of relevance even though the practices returned by WikiHow represent good results but do not satisfy the main aspect of the query.

We can thus conclude that our method manages to extract for each request the best practice in terms of overall consistency with the subject sought and also groups together the most important steps that each process must follow to achieve its objective.

5. CONCLUSION AND PERSPECTIVE

In this article we have presented a new method to conceptualize good practices and extract the best practice for a searched request. We have demonstrated the feasibility of our method on real knowledge extracted from the WikiHow sharing site. The use of graphs to model PK has proven to be very advantageous in terms of flexibility but also regarding the possibilities offered by such a formalism to identify the importance of the paths representing the best practices thanks to the different measures of centrality of the graphs.

The results of the experiment obtained demonstrated the superiority of the Word2Vec model for the research of practices similar to a query. We found that this metric manages to capture the high similarity scores compared to the similarity measures WMD, and TF-IDF and also that the results returned by our method were more relevant. We have also observed that DBSCAN is sensitive to the choice of the metric used, the silhouette scores obtained for the different lines of codes have shown that the clustering used is better than the others. It is difficult to evaluate a best practice among others, especially since we do not have a pre-established model but by comparing the best practice extracted by our method and the best practices returned in the article at the top of the site's list. We clearly see the relevance of the results of our method compared to other WikiHow results.

As a perspective, we plan to extend the search for good practices for a query, not dedicated only to the sentences describing the practices, but also to the steps they include, in order to identify the knowledge conveyed in practice as a whole for capturing an even more relevant know-how.

REFERENCES

[1] Lave, J., Wenger, E. (1991). Situated Learning: Legitimate Peripheral Participation. Cambridge University Press. http://dx.doi.org/10.1017/CBO9780511815355

[2] Fayyad, U., Piatetsky-Shapiro, G., Smyth, P. (1996). From data mining to knowledge discovery in databases. AI Magazine, 17(3): 37-37. https://doi.org/10.1609/aimag.v17i3.1230

[3] Tenorth, M., Klank, U., Pangercic, D., Beetz, M. (2011). Web-enabled robots. IEEE Robotics & Automation Magazine, 18(2): 58-68. http://dx.doi.org/10.1109/MRA.2011.940993

[4] Chu, C.X., Tandon, N., Weikum, G. (2017). Distilling task knowledge from how-to communities. In Proceedings of the 26th International Conference on World Wide Web, pp. 805-814. https://doi.org/10.1145/3038912.3052715

[5] Kiddon, C., Ponnuraj, G.T., Zettlemoyer, L., Choi, Y. (2015). Mise en place: Unsupervised interpretation of

instructional recipes. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 982-992. https://doi.org/10.18653/v1/D15-1114

[6] Yang, S., Zou, L., Wang, Z., Yan, J., Wen, J.R. (2017). Efficiently answering technical questions-A knowledge graph approach. In Thirty-First AAAI Conference on Artificial Intelligence, pp. 3111-3118. http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14576.

[7] Park, H., Motahari Nezhad, H.R. (2018). Learning procedures from text: Codifying how-to procedures in deep neural networks. In Companion Proceedings of the The Web Conference, pp. 351-358. https://doi.org/10.1145/3184558.3186347

[8] Feng, W., Zhuo, H.H., Kambhampati, S. (2018). Extracting action sequences from texts based on deep reinforcement learning. arXiv preprint arXiv:1803.02632. https://doi.org/10.24963/ijcai.2018/565

[9] Gupta, A., Khosla, A., Singh, G., Dasgupta, G. (2018). Mining procedures from technical support documents. arXiv preprint arXiv:1805.09780. https://arxiv.org/abs/1805.09780.

[10] Jung, Y., Ryu, J., Kim, K.M., Myaeng, S.H. (2010). Automatic construction of a large-scale situation ontology by mining how-to instructions from the web. Web Semantics: Science, Services and Agents on the World Wide Web, 8(2-3): 110-124. https://doi.org/10.1016/j.websem.2010.04.006

[11] Stanford NLP Group. (2020). The Stanford Natural Language Processing Group. https://nlp.stanford.edu/software/lex-parser.

[12] Etzioni, O., Fader, A., Christensen, J., Soderland, S. (2011). Open information extraction: The second generation. In Twenty-Second International Joint Conference on Artificial Intelligence, pp. 3-10. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-012

[13] Noura, M., Gyrard, A., Heil, S., Gaedke, M. (2019). Automatic knowledge extraction to build semantic web of things applications. IEEE Internet of Things Journal, 6(5): 8447-8454. https://doi.org/10.1109/JIOT.2019.2918327

[14] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems.

[15] Ester, M., Kriegel, H.P., Sander, J., Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In KDD, 96(34): 226-231.

[16] Regneri, M., Koller, A., Pinkal, M. (2010). Learning script knowledge with web experiments. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp. 979-988. https://aclanthology.org/P10-1100.pdf.

[17] Brin, S., Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems, 30(1-7): 107-117. https://doi.org/10.1016/S0169-7552(98)00110-X

[18] Glez-Peña, D., Lourenço, A., López-Fernández, H., Reboiro-Jato, M., Fdez-Riverola, F. (2014). Web scraping technologies in an API world. Briefings in Bioinformatics, 15(5): 788-797. https://doi.org/10.1093/bib/bbt026

[19] Kusner, M., Sun, Y., Kolkin, N., Weinberger, K. (2015). From word embeddings to document distances. In International Conference on Machine Learning, pp. 957-966.

[20] Google. Google archive. https://code.google.com/archive/p/word2vec.

[21] Garbade, M.J. (2018). A quick introduction to text summarization in machine learning. Towards Data Science. https://towardsdatascience.com/a-quick-introduction-to-text-summarization-in-machine-learning-3d27ccf18a9f.

[22] Gudivada Venkat, N., Rao, C.R. (2018). Computational analysis and understanding of natural languages: Principles. Methods and Applications, 38: 317-328.

[23] Wasserman, S., Faust, K. (1994). Social Network Analysis: Methods and Applications. Cambridge University Press. http://dx.doi.org/10.1017/CBO9780511815478

[24] Freeman, L.C. (1978). Centrality in social networks conceptual clarification. Social Networks, 1(3): 215-239. https://doi.org/10.1016/0378-8733(78)90021-7

[25] WikiHow. About wikiHow. https://www.wikihow.com/wikiHow:About-wikiHow.

[26] Python. Python Software Foundation. https://pypi.org/project.

[27] Chowdhury, G.G. (2010). Introduction to Modern Information Retrieval. Facet Publishing.

[28] Rousseeuw, P.J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20: 53-65. https://doi.org/10.1016/0377-0427(87)90125-7

[29] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1(14): 281-297.