



HAL
open science

A Cognitive Agent Framework in Information Retrieval: Using User Beliefs to Customize Results

Dima El Zein, Célia da Costa Pereira

► To cite this version:

Dima El Zein, Célia da Costa Pereira. A Cognitive Agent Framework in Information Retrieval: Using User Beliefs to Customize Results. The 23rd International Conference on Principles and Practice of Multi-Agent Systems, Nov 2020, Nagoya, Japan. 10.1007/978-3-030-69322-0_21 . hal-03029127

HAL Id: hal-03029127

<https://hal.science/hal-03029127>

Submitted on 27 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Cognitive Agent Framework in Information Retrieval : Using User Beliefs to Customize Results

Dima El Zein and Célia da Costa Pereira

Université Côte d’Azur
CNRS, I3S, UMR 7271, France
{elzein@i3s.unice.fr;Celia.DA-COSTA-PEREIRA@univ-cotedazur.fr}

Abstract. This paper presents a framework for a cognitive agent in information retrieval that customizes the list of returned documents based on what the agent believes about the user knowledge. Throughout the interactions between the agent and the user, the agent builds its beliefs by extracting the content of the documents examined by the user. The agent’s belief base consists of “simple beliefs” represented by the document’s keywords as well as “contextual rules” that allow the agent to derive new beliefs about the user knowledge. The agent is therefore able to compare its own beliefs about the user knowledge with the knowledge conveyed by a given document, and thus understand if the document really contains useful information for the user or not. Finally, in case of inconsistency, the agent revises its belief base to restore consistency.

Keywords: Cognitive Agent · Information Retrieval · Knowledge Extraction · Belief Revision.

1 Introduction and Motivation

The aim of Information Retrieval (IR) systems is to provide the user with *relevant* information for a given query. While in the traditional information retrieval approaches, relevance consists of matching the documents with the topics of the user query, more recent systems consider personalizing the returned results according to the user’s interests. To this aim, they use, for example, information about *query logs*, *click-through*, *time spent examining the document*, or the explicit and/or implicit feedback from the users. In recent decades, relevance has also been viewed as a multi-faceted concept and several criteria or dimensions have been considered to assess its value [1,2].

We believe that users can be considered as cognitive agents [3] having their own beliefs and knowledge about the world. They try to fulfill needs for information by requesting queries and acquire new information by examining the results. Taking into account the cognitive components of the user in information search engines has recently been set as one of the “major challenges” by the IR community [4]. However, many years ago, Maron already stressed in [5], that

a document relevance should consider the user’s prior knowledge or “cognitive map”. He defined the value of a retrieved document as the extent to which its content helps the user fill a gap in his/her cognitive map.

More recently, Cholvy *et al.* [6] proposed 3 definitions for measuring the *usefulness* of information for a cognitive agent. The definitions are based on the fact that a user requesting information needs it to achieve a goal. Therefore, they consider as useful, any piece of information allowing to decrease the information gap to achieve the goal. However, their proposal is adapted to a static situation in which the agent’s beliefs do not need to be updated after the arrival of new information.

In this paper, when returning results to the users, we are aiming to consider the cognitive attitudes of the users, in particular their knowledge. We propose a framework for Information Retrieval systems which is run by cognitive rule-based agent. The agent has some beliefs about the user knowledge that are extracted from the content of the documents examined by the user. The content extraction is done using the Rapid Extraction Keyword Extraction (RAKE) [7] method. A belief revision operator is incorporated into the model to ensure that the agent’s beliefs about the user are consistent and not contradictory.

2 Rule-Based Agents

A Rule-based agent consists of facts (ground literals) and rules (Horn clauses). Here, rules and facts are “treated” as beliefs. The facts represent information that the agent has currently obtained about its environment. They might change over time as a result of the addition/deletion of other facts from the agent’s beliefs due to: (i) new information (dynamic environment) (ii) the rule’s reasoning process itself. The rules are relationships between facts, and will be used to derive new beliefs (*derived beliefs*) from the agent’s existing beliefs.

The agent’s beliefs are then represented in predicate logic, in the form of literals and Horn clause rules. The rules have the form $\alpha_1 \& \alpha_2 \dots \& \alpha_n \rightarrow \beta$ where $\alpha_1, \alpha_2, \dots, \alpha_n (n \geq 1), \beta$ are literals. β is called the derived belief, and each α_i is a premise of the rule. The $\&$ represents the logical *and* operator.

3 Belief Revision

3.1 AGM Belief Revision

Belief revision is the process of modifying the belief base to maintain its consistency whenever new information becomes available. The AGM belief revision theory [8] defines postulates that a rational agent should satisfy when performing belief revision. In such theory, a belief base is closed under logical consequence. We consider a belief base K and a new piece of information α . K is inconsistent, when both α and $\neg\alpha$ are in $Cn(K)$, or $Cn(K) = \perp$, or both α and $\neg\alpha$ are logical consequences of K . Three operators are considered: *Expansion* $K + \alpha$: adds a new belief α that does not contradict with the existing beliefs. *Contraction* $K \div$

α : removes a belief α and all other beliefs that logically imply/entail it. *Revision* $K * \alpha$: adds a belief α as long as it does not cause contradiction in K . If the addition will cause inconsistencies in K , the revision operation starts by minimal changes in K to make it consistent with α , then adds α . In particular, if the agent has to contract a belief α , it does not contract other beliefs that derived α , as long they are consistent with the remaining beliefs (*minimal change*) – *coherence approach* [9].

The AGM framework comprises sets of postulates to be respected by these operators to ensure consistent and minimal belief revision.

On another side, the Reason-maintenance belief revision approach considers tracking dependencies between beliefs, so that the reason(s) for believing in a belief α can be traced. When α should be given up, the agent must ensure that α is no longer derivable and give up believing the things that derived it.

3.2 Partial Entrenchment Ranking

A belief is gradual and an agent might have beliefs more entrenched (or accepted) than others. Williams [10] have proposed a quantitative approach for the AGM framework, by developing finite partial entrenchment rankings to represent epistemic entrenchment – a piece of information is labelled by a degree of confidence denoting how strongly we believe it. The epistemic entrenchment [11] captures the notions of significance, firmness, or defeasibility of beliefs.

Epistemic entrenchment relations induce preference orderings of beliefs according to the importance of these beliefs in the face of change. If inconsistency arises during a belief revision operation, the least significant beliefs (i.e., beliefs with the lowest entrenchment degree) are given up in order to restore consistency.

3.3 Tracking Beliefs and preferences

We present in this section the approach proposed by Alechina *et al.* [12] to track beliefs and calculate their preference for rule-based agents. The core of their work was a proposition for belief revision and contraction for such agents which are a synthesis of AGM and reason maintenance styles. The details of their algorithms will not be described here as they do not fit the purpose of the paper. The authors considered preferences on beliefs, assuming that a user might prefer some beliefs over others. Those preferences are used to decide which belief(s) should be removed to restore consistency.

The dependency between beliefs is considered as follows. For every fired rule instance, a Justification J will record: (i) a belief α , which corresponds to the derived belief and (ii) a *support list*, s , which contains the premises of the rules (contextual beliefs of a plan used to derive α). The dependency information of a belief had the form of two lists: *dependencies* and *justifications*. A *dependencies list* records the justifications of a belief, and a *justifications list* contains all the Justifications where the belief is a member of a support.

They define preferences using a notion of quality associated with justifications, assuming the quality of a justification is represented by non-negative integers in the range $0, \dots, m$, where the value of 0 means lowest quality and m means the highest quality. The lower the value, the least the quality.

Definition 1. *The preference value of a belief α , $p(\alpha)$, is equal to that of its highest quality justification.*

$$p(\alpha) = \max\{qual(J_0), \dots, qual(J_n)\} \quad (1)$$

Definition 2. *The quality of justification J , $qual(J)$, is equal to the preference of the least preferred belief in its support list.*

$$qual(J) = \min\{p(\alpha) : \alpha \in \text{support of } J\} \quad (2)$$

Literals with no supports (or empty support) are assigned the lowest quality. An *a priori* quality is assigned to each justification with empty support.

4 Knowledge Extraction from documents - RAKE

The RAKE (Rapid Automatic Keyword Extraction) [7] is an unsupervised, domain-independent, and language-independent method used for extracting keywords from documents.

RAKE starts its extraction on a document by parsing the text into a set of candidate keywords. First, the document text is split into an array of words by specified word delimiters. This array is then split into sequences of contiguous words at phrase delimiters and stop word positions. Words within a sequence are assigned the same position in the text and together are considered a candidate keyword. After every candidate keyword is identified and the graph of word co-occurrences is complete, a score is calculated for each candidate keyword. The method evaluates several metrics for calculating word scores, based on the degree and frequency of word vertices in the graph: (1) word frequency $freq(w)$, (2) word degree $deg(w)$, and (3) ratio of degree to frequency $deg(w)/freq(w)$.

RAKE also identifies keywords that contain interior stop words such as *axis of evil*. To find these, it looks for pairs of keywords that adjoin one another at least twice in the same document and in the same order. A new candidate keyword is then created as a combination of those keywords and their interior stop words. The score for the new keyword is the sum of its member keyword scores. After candidate keywords are scored, the top one-third scoring candidates are selected as keywords for the document.

5 Proposed Framework

Our framework proposes a rule-based agent for Information Retrieval systems that uses its cognitive abilities to acquire beliefs about its user's knowledge to

customise the returned results. When the IR agent has α in its belief base, it believes that the user knows that α is true. If the belief base contains $\neg\alpha$, then the agent believes the user knows that α is not true. When neither α nor $\neg\alpha$ is in the belief base, the agent believes neither the user knows α is true nor the user knows that α is false.

The user knowledge is extracted from the documents he/she has examined and the rules are acquired through the integration with external entities like semantic entities, ontology or knowledge mining tools. We represent the contextual rules by Horn clause rules and consider them static. The origin of the rules is not discussed in this paper; this is left for future work.

As belief is gradual, we adopt the partial entrenchment approach briefly discussed in Section 3.2 to represent the degrees of the derived beliefs. The entrenchment degree for a derived belief will be calculated based on the premises of the rule that derived it. The degrees of the extracted beliefs instead, will be calculated based on the score of the corresponding keywords in the document. The beliefs can be revised and the relations between them are tracked as proposed by the Alechina’s approach and discussed in Section 3.3, i.e. by using the concept of *beliefs* and *justifications* with their related *preference degree value* and *quality of justification*.

Finally, we employ the agent beliefs to return filtered results to the user.

5.1 Model Architecture

In this section, we present a client-side web search agent implementing the methods discussed in the previous sections to personalize search while learning about the user knowledge. The agent can be a browser plugin acting as a proxy for web search engines or a filter built on top of existing search applications.

By running the agent on the client-side, no information about the user behaviour and activity is released to the outside as they all reside on his/her computer. Also, in terms of server scalability and efficiency, the model distributes computation and storage among its users rather than centralising them.

The framework has 3 main modules: (1) rule-based module, modeling the beliefs and rules. It performs inference reasoning about the user’s knowledge and revise its beliefs if needed, to maintain consistency. (2) Knowledge extractor module, extracting knowledge from the documents examined by the user. (3) Result filtering module; selecting the “useful” documents to be proposed to the user. These modules work together, as follows, for every submitted query: (i) the system sends the query to the search engine/application, (ii) receives the result from the search engine, and (iii) filters them by excluding those that are not useful to the user, according to the user knowledge. After the user has clicked on the results, the agent updates its beliefs about the knowledge supposed to be acquired by the user. Finally, a reasoning cycle is performed to run the applicable rules and revise the beliefs as needed.

5.2 Extracting Knowledge and Belief Entrenchment

The agent belief base is composed of the beliefs it learns about its user and of the contextual rules it owns. When the user reads a document d , the agent learns the user has acquired some knowledge about its content; so it uses some knowledge extraction/NLP tool to extract the representation of the document content. We use the RAKE method [7] as an easy and understandable method, to extract the keywords representing the document and calculate their related scores. The beliefs resulting from this process are referred to *extracted beliefs*. As the keywords are associated to the agent beliefs, their related scores are associated to degrees of beliefs. We define the degree of a belief as follows:

Definition 3. *The degree of a belief α is the degree to which the agent believes the user is knowledgeable about α . It is represented by a decimal ranging between 0 and 1, where 0 means the lowest degree –the agent believes the user has absolutely no knowledge about α , and 1 means the highest degree –the agent believes the user has the maximum knowledge about α .*

Let us consider document $d = \{k_1, \dots, k_n\}$. We suppose that a keyword k_i is associated with an extracted belief b_j whose degree is calculated as follows:

$$\text{degree}(b_j) = \lambda \cdot \frac{\text{Rakescore}(k_i)}{\max_{k_j \in d} \text{Rakescore}(k_j)}. \quad (3)$$

In Equation 3 the score of an extracted RAKE keyword is first divided by the highest keyword score of the document: the score is hence normalized. It is then multiplied by an adjustment factor $\lambda \in [0, 1]$. The adjustment factor is used to weaken the magnitude of the scores: when λ is equal to 1, the “top score” keyword of all examined documents will be assumed to have the maximal degree 1, which means that the agent assumes the user has the maximal knowledge about this keyword. The adjustment factor may vary based on different characteristics such as the source of the document, for example.

A *derived belief* is the belief resulting from firing the agent’s rules. The dependency between the beliefs is tracked by the concept of justification and dependencies lists as per Alechina’s approach presented in Section 3.3. Our definition of degree of derived beliefs is inspired by their notion of preference. Indeed, we use Equation 1 to calculate the degree of a derived belief, and Equation 2 to calculate the quality of a justification.

5.3 Revising Beliefs

While the agent is acquiring more information about the user, it is adding beliefs to its belief base. The beliefs might be new, already existing, or contradicting with the existing ones; that calls for the need of revising beliefs to ensure the belief base is consistent. The belief revision approach we have adopted here combines the AGM and reason-maintenance styles while tracking the beliefs.

In case of contradiction between a pair of beliefs, we give preference to the more entrenched one, i.e. the belief with the highest degree. Our model does not

contract a belief α unless a more preferred contradictory belief $\neg\alpha$ was added. When contracting a belief α , we don't see a need to contract beliefs that derived α : when the rule deriving α tries to add it again, the addition will be discarded because it will be faced by $\neg\alpha$ that is more preferred. In other terms, we contract the belief in question with its related justification(s), without contracting neither the rule's premises nor the rule itself. For example, consider an agent's belief base:

$$B = \{astronomy[0.8], astronomy \rightarrow planet(pluto)\}.$$

The agent reasoning will result in the execution of the rule and therefore adding $planet(pluto)$ with an entrenchment degree of 0.8. Suppose the agent was later informed that the agent believes pluto is not a planet, assuming the degree of entrenchment of the related belief is 0.9. The agent attempts to add $\neg planet(pluto)[0.9]$ in its belief base. Upon revising the beliefs, $planet(pluto)$ is contracted and $\neg planet(pluto)$ is added, since the belief with the higher degree is given priority. However, when $planet(pluto)$ is removed, the fact that the user knows about astronomy does not change: the belief $astronomy$ is not contracted. The related rule remains in the belief base as well.

5.4 Document Similarity and Result Filtering

The filtering process will be dependent on the similarity between the agent's current beliefs and the documents to be proposed to the user.

We propose a similarity function, $Sim(B, d)$, that calculates this similarity measure based on the degrees of the intersected beliefs and the document knowledge. The formula is inspired by the similarity function proposed by Lau *et al.* in [13]. We consider a set of beliefs $B = \{b_1, b_2, \dots, b_m\}$ and a document $d = \{k_1, k_2, \dots, k_n\}$ characterized by a set of keywords representing the content. Let us consider S , the set of keywords appearing both in d and in B defined by $S = \{k_i \in d : extent(B, k_i) > 0 \vee extent(B, \neg k_i) > 0\}$.

$$Sim(B, d) = \begin{cases} \frac{\max\{\sum_{k_i \in d} [extent(B, k_i) - extent(B, \neg k_i)], 0\}}{|S|} & \text{if } |S| \neq 0 \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

The $extent(B, k) = degree(k)$, if $k \in B$; and 0 otherwise. The $degree(k)$ is calculated as discussed in Section 5.2. The similarity formula "rewards" the documents containing keywords that are in the set B .

We set a cutoff value γ that allows to decide whether the knowledge inside a document is similar to a set of beliefs or not. The filter is used according to the intended application: when the purpose of framework is to reinforce the user knowledge, then documents that are "near" the agent beliefs will be returned. The documents having a similarity score greater than the cutoff will be returned as the final results to the user. Contrarily, when the framework is employed for novelty purposes, the documents having similarity below the cutoff will be returned.

6 Conclusion and Future Work

In this paper, we proposed an innovative framework for a rule-based information retrieval agent which relies on its cognitive abilities to learn about the user's knowledge, and propose new/relevant documents accordingly.

The components of the framework are: (1) rule-based module, modeling the agent's beliefs and rules. It performs inference reasoning about the user's knowledge and revise its beliefs if needed, to maintain consistency. (2) Knowledge extractor module, extracting knowledge from the documents examined by the user. (3) Result filtering module, selecting the "useful" documents to be proposed to the user. The process of acquisition of information is session and time independent. Another advantage of the proposed framework is that it can be built on top of existing non-cognitive search engines, applications, or open-source libraries.

As for future work, we plan to take into account the confidence in the sources of the documents, which will probably affect the degree of entrenchment of a belief. Another possible extension could be to integrate some semantic analysis in order to deal with synonyms.

References

1. G. Pasi, G. Bordogna, and R. Villa, "A multi-criteria content-based filtering system," 01 2007, pp. 775–776.
2. C. da Costa Pereira, M. Dragoni, and G. Pasi, "Multidimensional relevance: A new aggregation criterion," 04 2009, pp. 264–275.
3. M. Móra, G. Lopes, R. Maria Vicari, and H. Coelho, "Bdi models and systems: Bridging the gap." 01 1998, pp. 11–27.
4. J. S. Culpepper, F. Diaz, and M. D. Smucker, "Research frontiers in information retrieval: Report from the third strategic workshop on information retrieval in lorne (SWIRL 2018)," *SIGIR Forum*, vol. 52, no. 1, pp. 34–90, 2018.
5. Maron, "Mechanized documentation: The logic behind a probabilistic interpretation," National Bureau of Standards Miscellaneous Publications, 1964.
6. L. Cholvy and C. da Costa Pereira, *Usefulness of Information for Goal Achievement*, 10 2019, pp. 123–137.
7. S. Rose, D. Engel, N. Cramer, and W. Cowley, *Automatic Keyword Extraction from Individual Documents*, 03 2010, pp. 1 – 20.
8. C. E. Alchourrón, P. Gärdenfors, and D. Makinson, "On the logic of theory change: Partial meet contraction and revision functions," *J. Symb. Log.*, vol. 50, no. 2, pp. 510–530, 1985.
9. P. Gärdenfors, *Belief revision: An introduction*, ser. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1992, pp. 1–28.
10. M.-A. Williams, "Iterated theory base change: A computational model." 01 1995, pp. 1541–1549.
11. P. Gärdenfors and D. Makinson, "Revisions of knowledge systems using epistemic entrenchment," 03 1988, pp. 83–95.
12. N. Alechina, M. Jago, and B. Logan, "Preference-based belief revision for rule-based agents," *Synthese*, vol. 165, pp. 159–177, 11 2008.
13. R. Lau, P. Bruza, and D. Song, "Belief revision for adaptive information retrieval," 07 2004.