ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240629

The Form and the Content: Non-Monotonic Reasoning with Syntactic Contextual Filtering

Florence Dupin de Saint-Cyr [a,*] **and Pierre Bisquert** [b,c]

[a]Lab-STICC_COMMEDIA, CNRS, Brest, France.
[b]IATE, Univ Montpellier, INRAE, Institut Agro, Montpellier, France
[c]Inria, LIRMM, Univ Montpellier, CNRS, France

Abstract. In order to avoid ambiguity and be efficient, the context in which a query is made can help to better target the relevant pieces of information from the knowledge base to be processed by the inference system. In this paper, we are interested in the notion of *dynamical compartmentalization* where the knowledge base that will be used for reasoning is dynamically extracted from the original base. Compartmentalization is a selection of a sub-base which is done according to a function, called *refiner*, and depending on this function some properties are satisfied. We introduce a particular *syntactic refiner* that uses a similarity symbol-based distance between a context (a multiset of variable symbols) and a formula of a knowledge base. We prove that the inference operator based on this refiner, called *contextual inference*, satisfies a series of desirable axioms

1 Introduction

With the advent of big data came the need to integrate more and more data sources to ensure that no aspect of a question, or query, is missed. This however raises some important issues regarding privacy and rights to access and process such data. A potential answer to this would be to tailor databases for particular queries and/or users on a use-case basis, which would ensure the notion of *compartmentalization* at the cost of being tedious and having a lot of data redundancy, and its corollary: the expensive question of storage. Another way of managing this heterogeneous knowledge is to compartmentalize it only when it is accessed, as proposed in this paper.

In this article, our main goal is both to handle inconsistency and non-monotonic reasoning together with efficiency. More precisely, we consider that a rational agent attempts to answer queries and evaluate arguments based on what it knows. The idea is that its knowledge results from a blind integration of everything that has been heard, which can lead to inconsistencies. By answering a query, we mean checking whether a formula is entailed by the knowledge base in a given context. By assessing an argument we mean checking whether its set of premises entails its claim in this context.

Even if the knowledge base is inconsistent it can still be exploited by selecting consistent subbases that are contextually relevant, mimicking the human brain that activates pieces of information based on their relationship to a context (see e.g. chunks in cognitive theories such as ACT-R [2]). We propose to create the compartmentalization on the fly when answering a query or assessing an argument: this involves the notion of *dynamical subbase selection* where the part of the knowledge base that will be used to reason and answer a query is dynamically extracted from the original base. This selection is done according to a generic function, called *refiner*, and depending on this function some properties will be satisfied.

In particular, we are interested in the notion of *syntactic refiner* that will select the subbase based on the symbols appearing in the query. Intuitively, such a selection is easier to do based on syntactic criteria (compared with a full-blown semantic selection) but it may miss some relevant formulas. Moreover, to the best of our knowledge, there are no studies that attempt to account for the impact of the way a formula is written, while retaining the same semantics. Nevertheless, the fact that particular symbols are repeated, or that seemingly irrelevant symbols appear, can be considered significant, as is the case in marketing or politics for example, where the language used is strategically chosen to have a greater impact. The question is: what conditions must the refiner meet to ensure that the syntactic selection of the subbase allows for sound and complete reasoning?

After stating the notations used in the paper, Section 2 recalls the basics of non-monotonic reasoning. Our main contribution is presented in Section 3, where we define the notions of generic refiner relative to a context, on which we base the definition of a new type of inference operator. Next, we propose a specific refiner based on syntactic similarity, we prove that given a context, this inference is rational but in order to better characterize the impact of context on reasoning, we propose three desirable axioms related to these operators as well as two optional axioms about the sensitivity to the syntax. We conclude by discussing future directions.

Notations

We consider a propositional language $\mathcal{L}$ containing formulas denoted by lower case Greek letters, based on a vocabulary $\mathcal{V}$ of variable symbols denoted by Latin lower case letters and containing the two constant symbols $\bot$, $\top$ for denoting contradiction and tautology respectively. Negation, conjunction, disjunction, material implication, equivalence and classical inference are denoted respectively by $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\equiv$ and $\models$. Let $K \subseteq \mathcal{L}$ be a *finite* set, not necessarily consistent, of consistent formulas of $\mathcal{L}$ representing the knowledge base of an agent. The formulas of K can be ranked according to their importance by a complete pre-order on K denoted by $\succeq$, resulting in a prioritized knowledge base denoted by $P = (K, \succeq)$. The pre-order $\succeq$ is supposed to be given (in case no importance order has been given, all formulas of K are considered as equally important). For any for-

* Corresponding author email: dupin@irit.fr

mulas $\alpha, \beta \in K$, $\alpha \succeq \beta$ is read α *has priority over* (is at least as plausible as) β. Since K is finite, $\succeq$ can be translated into a ranking of the formulas of K based on the equivalence classes of $\simeq$ induced by $\succeq$ (with $a \simeq b$ iff $a \succeq b$ and $b \succeq a$). Given $\succeq$ and K there is a unique index n corresponding to the set of formulas with the lowest priority (denoted by K_n), n is called the lowest priority rank. Given $\succeq$, any subset E of K can be decomposed accordingly to the ranks of its formulas, formulas of highest priority of E are denoted by E_1 and more generally the subsets of formulas of rank i are denoted by E_i, $E_i = E \cap K_i$. With these conventions, a prioritized knowledge base $(K, \succeq)$ is equivalently represented by the tuple $(K_1, \dots, K_n)$. The strict order induced by $\succeq$ is denoted by $\succ$ (with $a \succ b$ iff $a \succeq b$ and not $b \succeq a$).

We associate each formula φ with the multi-set $\mathrm{ms}(\varphi)$ of variables symbols occurring in it: it is a pair (V_φ, m_φ) where $V_\varphi \subseteq \mathcal{V}$ and $m_\varphi : \mathcal{V} \to \mathbb{N}$ is the multiplicity of the symbols in the multiset. For sake of shortness, a multiset is also represented by the tuple of its symbols (possibly repeated) in alphabetical order, the empty multiset is denoted by (). For instance, the formula $\varphi = p \vee (\neg p \wedge f)$ is associated with the multiset $\mathrm{ms}(\varphi) = (\{p, f\}, m_\varphi)$ with m_φ such that $m_\varphi(x) = 0$ for any symbol $x \in \mathcal{V} \setminus \{p, f\}$ and $m_\varphi(p) = 2$ and $m_\varphi(f) = 1$. This multiset can also be denoted by (f, p, p). As a multiset intersection, $\sqcap$ gives the multi-set $(A \cap B, m)$ where $m(x) = \min(m_A(x), m_B(x))$ for all $x \in \mathcal{V}$. When there is no ambiguity with the operation performed on it, *we abuse notations by writing the formula*, e.g. φ, *instead of its multiset*: $\mathrm{ms}(\varphi)$. The symbol $\sqcup$ denotes the union between multi-sets, it is the multiset $(A \cup B, m)$ where $m(x) = \max(m_A(x), m_B(x))$. The symbols $\in$ and $\subseteq$ are naturally extended to multi-sets. $|M|$ denotes the cardinality of the multiset M: it is the sum of the multiplicities of its symbols. The set of all multi-sets of symbols in $\mathcal{V}$ is denoted by $\mathcal{MS}$.

2 Basics about non-monotonic reasoning

Inconsistency can arise from different situations: experts or sensors providing conflicting data (differing opinions, incompatible measurements), rules applicable in a context with contradictory conclusions, incompatible goals of several agents, etc. In the presence of inconsistency, classical logical deduction is unusable (it deduces any formula and its opposite). Many approaches have been proposed to handle this significant problem. In this paper, we focus on "syntax-based approaches"[1] that inherit from [19] who first introduced the approaches based on maximum (for set-inclusion) consistent subbases of the knowledge base. In this kind of approaches, each formula of the knowledge base is considered as an independent piece of information. The solutions proposed in this domain are introducing *non-monotonic inference relations* defined by selecting "preferred"[2] subbases of the knowledge base on which classical inference is applied. This preference can be based on an existing ordering of the knowledge base as in [5]. In the following we adopt the conventions used in [3] for recalling different classical non-monotonic inference relations. In these definitions, B, B', K are subsets of formulas of $\mathcal{L}$ and $\alpha, \beta, \alpha', \beta', \gamma$ are formulas of $\mathcal{L}$.

Definition 1. *$B \subseteq K$ is an* inclusion maximal α-consistent *subbase of K iff $\{\alpha\} \cup B$ is consistent and there is no $B' \subseteq K$ s.t. $B' \supset B$ and $B' \cup \alpha$ is consistent.*

Note that inclusion maximal consistent subsets were later called maximal satisfiable subsets (MSS) in [13]. The inference based on inclusion maximal consistent subsets is defined below.

Definition 2 (MSS inference)**.** *$\alpha \mid\!\sim_K^{\mathrm{MSS}} \beta$ iff for all inclusion-maximal α-consistent subbase B of K, $B \cup \{\alpha\} \models \beta$.*

Hereafter, $\mid\!\sim_K^{\mathrm{MSS}} \beta$ is a shortcut for $\top \mid\!\sim_K^{\mathrm{MSS}} \beta$. We can notice that MSS inference is non-monotonic, as illustrated on Example 1.

Example 1. *Let us consider the following (typical) knowledge base (expressing that penguins are birds that do not fly, birds fly and have wings): $K_1 = \{\varphi_1.p \to b, \varphi_2.p \to \neg f, \varphi_3.b \to f, \varphi_4.b \to w\}$, the reader can check that f can be inferred from $K_1 \cup \{b\}$: $\mid\!\sim_{K_1 \cup \{b\}}^{\mathrm{MSS}} f$ (since there is only one subbase maximally consistent with $\{b\}$: $K_1 \cup \{b\}$). However, in the presence of b and p this no longer holds since there are 3 subbases maximally consistent with $\{b, p\}$: $K_1 \cup \{b\}$, $\{b, p, \varphi_1, \varphi_2, \varphi_4\}$ and $\{b, p, \varphi_1, \varphi_3, \varphi_4\}$, among them the bird may fly or not: hence, it holds that $\not\mid\!\sim_{K_1 \cup \{b,p\}}^{\mathrm{MSS}} f$.*

2.1 System P

Non-monotonic inference relations have been particularly studied by Kraus, Lehmann and Magidor [10], these authors have proposed a set of inference rules called System P for augmenting an existing set of inferences with new inferences that should follow rationally from them.

Definition 3 (System P [10])**.** *An operator $\mid\!\sim$ satisfies System P if the following properties hold $\forall \alpha, \beta \in \mathcal{L}$:*

$\alpha \mid\!\sim \alpha$	*(Reflexivity)*
If $\alpha \equiv \alpha'$ *and* $\alpha \mid\!\sim \beta$ *then* $\alpha' \mid\!\sim \beta$	*(Left Logic. Equivalence)*
If $\beta \models \beta'$ *and* $\alpha \mid\!\sim \beta$ *then* $\alpha \mid\!\sim \beta'$	*(Right Weakening)*
If $\alpha \mid\!\sim \gamma$ *and* $\beta \mid\!\sim \gamma$ *then* $\alpha \vee \beta \mid\!\sim \gamma$	*(Or)*
If $\alpha \mid\!\sim \beta$ *and* $\alpha \mid\!\sim \gamma$ *then* $\alpha \wedge \beta \mid\!\sim \gamma$	*(Cautious Monotony)*
If $\alpha \wedge \beta \mid\!\sim \gamma$ *and* $\alpha \mid\!\sim \beta$ *then* $\alpha \mid\!\sim \gamma$	*(Cut)*

2.2 Prioritized inference

Apart from MSS inference, it is possible to build more refined non-monotonic inference relations by taking into account some priorities between formulas. Given a prioritized knowledge base $P = (K, \succeq)$: three orderings between subbases were introduced in [3], namely *best-out*, *incl* and *lex*, they allow us to compare subbases of K according to different criteria. Best-out ordering compares the priorities of the highest priority formulas which are out of the considered subbases, Incl ordering compares the sets of formulas of the two subbases strata by strata starting from the highest priority strata. Lex ordering is described below.

Such orderings on subbases enable us to select the best consistent subbases of a prioritized knowledge base. More precisely, an ordering o allows us to compare subbases of a prioritized knowledge base and to select the preferred ones according to o, called *o-preferred* subbases. The most common approach called *strong inference*, which is chosen in this paper, is to define that α non-monotonically infers β given the knowledge base K iff *all* the max α-consistent o-preferred

[1] This expression comes from [16] who defined syntax-based revision procedures where two semantically equivalent knowledge bases may result in non-semantically equivalent revisions. Note that the word syntax is used here in a very restricted sense: it concerns rather the way formulas are separated from each other than the way they are precisely written: e.g., in these approaches $K = \{a, b\}$ is distinguished from $K = \{a \wedge b\}$ but not from $K = \{a \wedge (c \vee \top), b\}$.

[2] Here as in the related literature, the word "preferred" is abusively used to qualify the plausibility of the pieces of information for the reasoning process (it is not related to a user's taste, i.e. a user utility function as in decision theory, otherwise there could be cases of wishful thinking).

subbases of K, together with α, classically entail β. Note that other inference principles exist [18], such as *weak* principle based on the existence of at least one max α-consistent o-preferred subbase entailing β, or *argumentative* principle which is weak inference s.t. no max α-consistent o-preferred subbase entails classically $\neg\beta$.

In the remainder of this work, we focus on Lex ordering [3, 12] defined below, which has the benefit of being the most refined ordering. However, our definitions could be used with any stratification-based selection function.

Definition 4 (Lex-preference)**.** *Given a prioritized knowledge base $P = (K_1, \ldots, K_n)$ with lowest priority rank n, given $A, B \subseteq K$, A is* Lex-equivalent *to B given P, denoted by $A \sim_P B$, iff $\forall i, |A_i| = |B_i|$. A is* strictly Lex-preferred *to B given P, denoted by $A \succ_P B$, iff $\exists k \in [1, n]$ s.t. $\begin{cases} |A_k| > |B_k| \text{ and} \\ \forall i < k, |A_i| = |B_i| \end{cases}$*

A is a Lex-preferred *α-consistent subbase of P if it is an α-consistent subbase of K s.t. any α-consistent subbase $B \subseteq K$ is s.t. $B \not\succ_P A$.*

Notation *$Lex(P, \alpha) = \{B \subseteq K \mid B$ Lex-preferred α-consistent subbase of $P\}$.*

We are now in position to recall Lexicographic entailment:

Definition 5 (Lex-entailment)**.** *Given a belief base $P = (K, \succeq)$ and two formulas α and β,*

$$\alpha \mathrel{|\!\sim}^{\text{lex}}_P \beta \text{ iff for any } B \in Lex(P, \alpha), B \cup \{\alpha\} \models \beta.$$

Note that [3] have shown that for any set $K \subset \mathcal{L}$ of formulas, $|\!\sim^{\text{MSS}}_K$ satisfies System P and for any prioritized base $P = (K, \succeq)$, $|\!\sim^{\text{lex}}_P$ satisfies System P.

Example 1 (continued): *Let us consider the following priorities among the formulas of K_1: $\varphi_1 \simeq_1 \varphi_2 \succ_1 \varphi_3 \simeq_1 \varphi_4$, leading to a prioritized base $P_1 = (K_1, \succeq_1)$.*

There is only one Lex-preferred subbase consistent with penguins: $Lex(P_1, p) = \{\{\varphi_1, \varphi_2, \varphi_4\}\}$, hence $p \mathrel{|\!\sim}^{\text{lex}}_{P_1} \neg f$. The reader can also check that $p \wedge b \mathrel{|\!\sim}^{\text{lex}}_{P_1} \neg f$.

What has also been shown in [3] is that from a prioritized knowledge base $P = (K, \succeq)$, it is possible to define a complete preorder on interpretations (based on the Lex-preference on the maximally consistent subbases they satisfy), and that Lex-entailment is a preferential entailment [14], that also satisfies the rule of Rational Monotony described in [7]:

$$\text{If } \alpha \mathrel{|\!\not\sim} \neg\beta \text{ and } \alpha \mathrel{|\!\sim} \gamma \text{ then } \alpha \wedge \beta \mathrel{|\!\sim} \gamma \qquad \text{(Rational Monotony)}$$

To sum up, from prioritized knowledge bases it is possible to define a rational non-monotonic inference relation but for this purpose a pre-ordering of the knowledge base is required: it may come from experts, but this information is not necessarily easy to obtain or is not consensual; it can be computed based on specificity notions (see next paragraph) but it is computationally costly.

2.3 Computing priorities: the System Z algorithm

In [17], Pearl defined an ordering, called "Z ordering", induced from a set of default rules. Indeed, in System Z, a *default rule* is a formula of the form $\alpha \rightsquigarrow \beta$ where α and β are propositional formulas of $\mathcal{L}$, and $\rightsquigarrow$ is a new connective, the intended meaning of the rule is "α generally entails β". More formally its interpretation is: the *most plausible*, according to a set of default rules Δ, models of the formula α satisfy β. The plausibility of interpretations given the set of defaults Δ are computed by considering each rule of Δ as a constraint on the ranking of interpretations, namely the rule $\alpha \rightsquigarrow \beta$ imposes that the interpretations satisfying $\alpha \wedge \beta$ are more plausible than the one satisfying $\alpha \wedge \neg\beta$ (the reader can refer to [4] for a reading of these rules in possibility theory). The Z ordering method, described in [17], is based on the tolerance notion between rules. More precisely, a rule $r = \alpha \rightsquigarrow \beta$ is tolerated by a set of n rules $R \subseteq \Delta$ iff $\alpha \wedge \beta \wedge \bigwedge_{\alpha_i \rightsquigarrow \beta_i \in R}(\neg\alpha_i \vee \beta_i)$ is consistent. The process starts by selecting the rules that are tolerated by Δ, they are attributed the level Z=0 and removed from Δ, then assign level Z=1 to the rules tolerated by all the remaining ones and so on. This process requires to practice $O(n^2)$ satisfiability tests in the worse case where $n = |\Delta|$.

Apart from the computational complexity for computing the Z ordering, there are two other limitations of the use of System Z: first, it is possible that a set of default rules does not admit a Z ordering, such defeasible set is called "inconsistent" in [6], second this ranking requires that the knowledge base is written under the form of default rules.

In what follows we propose an approach that refines an existing ordering (or creates one when it does not exist) without requiring extra information, this is done by filtering the formulas according to their contextual relevance. This relevance is based on the syntax by considering the symbols used in the formulas relatively to a given set of symbols representing the context. Using syntax can be very efficient since no call to a SAT solver is required, however it should be restricted in order to guarantee a rational behavior as we will see in Section 3.3. Moreover using only variable symbols to qualify the relevance of formulas to a given context can seem simplistic, but it opens the way to more realistic extensions based on additional knowledge of, for example, the lexical fields of the symbols.

3 Contextual inference

Non-monotonic inferential mechanisms, as the ones recalled above, use selection functions that are independent from the query to assess. In this paper, we are interested in defining a mechanism that takes into account a context (a multiset of propositional variable symbols in $\mathcal{V}$). For this purpose, we propose to reduce and reorder the knowledge base according to its relevance to a context, and then Lexicographic inference is used on this smaller prioritized knowledge base.

3.1 Generic definitions

In this section, we define a generic refiner λ which, given a knowledge base $P = (K, \succeq)$ and a context C, returns a new prioritized belief base.

Definition 6 (Refiner)**.** *Given a belief base $P = (K, \succeq)$ and a context $C \in \mathcal{MS}$, a* refiner *λ is a function such that:*

$$\lambda(P, C) = (K_C, \succeq_C)$$

where $K_C \subseteq K$ and $\succeq_C$ is a complete pre-order on K_C built from C and $\succeq$.

This notion of refining wrt a context C will be used to answer the question of whether "one can conclude about" a given formula in the context determined by C.

Definition 7 (Contextual inference, query)**.** *Given a belief base $P = (K, \succeq)$, two consistent formulas α and β, a context $C \in \mathcal{MS}$ and a* *refiner λ, the* contextual inference *based on λ, P and C is s.t.:*

$$\alpha \mid\!\sim^{\lambda}_{P,C} \beta \text{ iff for any } B \in Lex(\lambda(P,C),\alpha), B \cup \{\alpha\} \models \beta.$$

Moreover, when $\alpha = \top$, this inference is called query *about β in the context C, denoted by $\mid\!\sim^{\lambda}_{P,C} \beta$.*

In other words, $\alpha \mid\!\sim^{\lambda}_{P,C} \beta$ iff $\alpha \mid\!\sim^{\text{lex}}_{\lambda(P,C)} \beta$. In this work, given a knowledge base P, what ultimately motivates us are the cases where the context of the inference is determined by the query itself, i.e. $\mid\!\sim^{\lambda}_{P,\text{ms}(\beta)} \beta$.

Proposition 1. *Given a belief base P, a context C and a refiner λ, $\mid\!\sim^{\lambda}_{P,C}$ is a rational inference relation hence it satisfies Reflexivity, LLE, RW, CM, Cut and Rational monotony*

Proof. By definition, $\mid\!\sim^{\lambda}_{P,C}$ is a Lexicographic entailment which is a rational entailment [3]. □

Note that the previous proposition ensures that contextual inference satisfies System P and Rational monotony axioms, but they hold when the context C is fixed, i.e., C remains the same in the left and right parts of the if/then statements, for instance LLE becomes "For any $C \in \mathcal{MS}$ and any prioritized knowledge base P, $\forall \alpha, \alpha', \beta \in \mathcal{L}$, if $\alpha \equiv \alpha'$ and $\alpha \mid\!\sim^{\lambda}_{P,C} \beta$ then $\alpha' \mid\!\sim^{\lambda}_{P,C} \beta$".

Proposition 2 (One-way deduction theorem)**.** *Given a prioritized knowledge base $P = (K, \succeq)$, a refiner λ and a context $C \in \mathcal{MS}$, $\forall \alpha, \beta \in \mathcal{L}$, $\alpha \mid\!\sim^{\lambda}_{P,C} \beta$ implies $\top \mid\!\sim^{\lambda}_{P,C} \neg\alpha \vee \beta$.*

Proof. Theorem 3 of [20] establishes that any preferential entailment $\mid\!\sim$ is such that from $A \wedge B \mid\!\sim C$ we get $A \mid\!\sim B \rightarrow C$. □

Thanks to this result, querying $\neg\alpha \vee \beta$ (i.e. assessing $\mid\!\sim^{\lambda}_{P,\text{ms}(\alpha)\sqcup\text{ms}(\beta)} \neg\alpha \vee \beta$) can be done by checking whether $\alpha \mid\!\sim^{\lambda}_{P,\text{ms}(\alpha)\sqcup\text{ms}(\beta)} \beta$. Let us recall that if we were to accept the other way of the deduction theorem, i.e., from $A \mid\!\sim B \rightarrow C$ deduce $A \wedge B \mid\!\sim C$ then, due to Lemma 3 of [10], we would get monotonicity (from $\models A \rightarrow B$ and $B \mid\!\sim C$ deduce $A \mid\!\sim C$). Since monotonicity is not desirable here, Prop. 2 is only a one-way deduction theorem.

Remark 1. *Obviously refining P with a context C such that P is not modified amounts to using the classical lexicographic inference on the whole base P:*

$$\text{if } \lambda(P,C) = P \text{ then } \mid\!\sim^{\lambda}_{P,C} = \mid\!\sim^{\text{lex}}_{P}.$$

When the refiner operates (i.e., $\lambda(P,C) \neq P$), let us notice that $\mid\!\sim^{\lambda}_{P,\text{ms}(\beta)} \beta$ is neither a necessary nor a sufficient condition for $\mid\!\sim^{\text{lex}}_{P} \beta$, as shown in the following example. However, for the specific refiner defined in next section, we will provide conditions on the context (see Proposition 4) under which this contextual entailment conforms to lexicographic entailment.

Example 2. *Let us consider the following knowledge base with no priority: $P_2 = (K_2 = \{b, c, c \rightarrow a, a \rightarrow \neg b\}, K_2 \times K_2)$. Let us take a refiner λ_0 that selects formulas with at least one common symbol with the context and that keeps the same pre-order. The refinement in the context (b) is $P_{2(b)} = \lambda_0(P_2,(b)) = (K_{2(b)} = \{b, a \rightarrow \neg b\}, K_{2(b)} \times K_{2(b)})$. We get $K_{2(b)} \models b$ while there is one maximal (in cardinality) consistent subset of K_2, namely $\{c, c \rightarrow a, a \rightarrow \neg b\}$, that entails $\neg b$, i.e., $\mid\!\sim^{\lambda_0}_{P_2,(b)} b$ holds, but $\not\mid\!\sim^{\text{lex}}_{P_2} b$.*

Let us now consider a knowledge base that allows us to deduce b while after refining it with λ_0, it is no more the case. Let $P_3 = (K_3 = \{c, c \rightarrow a, a \rightarrow b\}, K_3 \times K_3)$, $P_{3(b)} = \lambda_0(P_3,(b)) = (K_{3(b)} = \{a \rightarrow b\}, K_{3(b)} \times K_{3(b)})$. We get that $\not\mid\!\sim^{\lambda_0}_{P_3,(b)} b$, but $\mid\!\sim^{\text{lex}}_{P_3} b$ holds.

Now that we know that contextual entailment is a rational inference relation given a fixed context, these examples show that by considering a "bad" refinement in a precise context, some crucial information may be lost allowing potentially undesirable inferences. This is why it is important to define some rational properties for the refiners. In the following section we propose a particular syntactic refiner before enunciating some desirable properties and checking whether they hold for this operator.

3.2 Syntactic similarity

In this section, we propose a new operator based on syntactic similarity: the idea is to obtain a behavior close to a cognitive activation process, i.e., first very relevant concepts come to the mind, then some others concepts that are related to the ones last activated, and so on. We choose to enforce that the first element that came to mind based on the context is considered more relevant than the one that came to the mind because of another element not directly related to the context. For this purpose we are going to define first the distance between two multisets then we define a distance path as a tuple of distances, where each edge relates multisets that have at least one common symbol. The shorter the path between two multisets the more relevant they are. In case of paths with the same number of edges the distance values along the paths are compared lexicographically.

Definition 8 (Distance between multisets)**.** *The distance between two multisets $C = (V, m)$ and $C' = (V', m')$ of $\mathcal{MS}$ is:*

$$d(C, C') = \sum_{s \in V \cup V'} |m(s) - m'(s)|.$$

Definition 9 (Distance path and multiset similarity)**.** *Given two distinct multisets $C, C' \in \mathcal{MS}$, a* path *from C to C' in a universe $U \subseteq \mathcal{MS}$ is a sequence of $n+1$ multisets $E_1, \ldots E_{n+1}$ with $n \geq 1$ such that $C = E_1$ and $C' = E_{n+1}$ and $\forall i \in [2, n]$ $E_i \in U$ and $\forall i \in [2, n+1]$, $E_{i-1} \sqcap E_i \neq ()$. When such a path exist the multisets are said* syntactically connected *(or s-connected) in U. The* distance path *associated to this path is the n-tuple $(d_1, d_2, \cdots, d_n)$ such that $d_i = d(E_{i-1}, E_i)$.*

Let $d = (d_1, \ldots, d_k)$ and $d' = (d'_1, \ldots, d'_l)$ be two distance paths where $k, l \geq 1$, d is lex-shorter *than d' iff $k < l$ or $k = l$ and $\exists i_0 \in [1, k]$ s.t. $\forall i \in [1, i_0]$, $d_i = d'_i$ and $d_{i_0+1} < d'_{i_0+1}$*

Given three multisets $C, C', C'' \in \mathcal{MS}$, C' is strictly syntactically more similar *to C in the universe $U \subset \mathcal{MS}$ than C'' is, denoted by $C' >^{\text{syn}}_{U,C} C''$, iff ($C' = C$ and $C'' \neq C$) or there is a distance path from C to C' in U lex-shorter than every distance path from C to C'' in U.*

In the following when U is omitted, it means that we consider the universe made of all the multisets of the formulas of the current knowledge base. The universe associated to a knowledge base K is $U_K = \bigcup_{\varphi \in K}\{\text{ms}(\varphi)\}$

Example 3. *Let us consider the universe U_1 of multisets:*
$C_1 = (a, a, b, c)$ $C_2 = (a, a, a, b)$ $C_3 = (a, b, c)$
$C_4 = (a, a, a, b, c)$ $C_5 = (c, d)$ $C_6 = (d, d, f)$

Figure 1 shows the graph associated to U_1. There are several distance paths in U_1 from C_1 to C_6: for instance (4,3) corresponding to (C_1, C_5, C_6) and (1,3,3) corresponding to (C_1, C_3, C_5, C_6).

The lex-shortest distance path from C_1 to C_6 in U_1 is $(4, 3)$. Let us compare the syntactic similarity of C_1 and C_2 to C_6, we have $C_1 >^{\text{syn}}_{U_1,C_6} C_2$ since there is a two edges path from C_1 to C_6 while all paths from C_2 to C_6 have at least 3 edges.

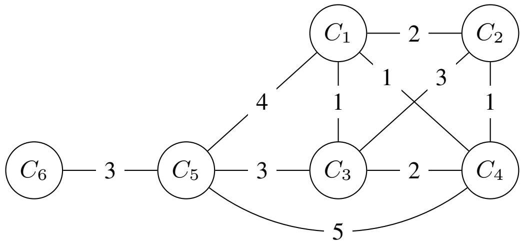

Figure 1. Paths between the six multisets of Example 3.

Definition 10 (λ_{syn} operator)**.** *The syntactic refiner, denoted by λ_{syn}, is defined as follows: given a prioritized knowledge base $P = (K, \succeq)$ of formulas and C a context*

$$\lambda_{\text{syn}}(P, C) = (K_C, \succeq_C)$$

with $K_C = \{\varphi \in K$ s.t. φ is s-connected to C in $U_K\}$ and $\succeq_C = \{(\alpha, \beta) \mid \alpha, \beta \in K_C$ and $\alpha >_C^{\text{syn}} \beta$ or ($\alpha \not>_C^{\text{syn}} \beta$ and $\beta \not>_C^{\text{syn}} \alpha$ and $\alpha \succeq \beta$)$\}$.

In other words, the λ_{syn} refiner selects first the most relevant formulas of the base K wrt a context C (according to $>_C^{\text{syn}}$) and, in case of equal relevance, the initial ordering ($\succeq$) of P is used. In the following the non-monotonic inference $\mid\!\sim^{\lambda_{\text{syn}}}$ based on the refiner λ_{syn} will be denoted by $\mid\!\sim^{\text{syn}}$.

Example 3 (continued): *Let us consider the formulas:*

φ_{11}.	$a \wedge (a \to b \vee c)$	φ_{12}.	$a \vee (\neg a \to \neg a \to b)$
φ_{13}.	$(a \wedge (b \to c))$	φ_{14}.	$a \vee (\neg a \to \neg a \to b \vee c)$
φ_{15}.	$c \wedge d$	φ_{16}.	$d \wedge (d \to f)$

Note that their respective associated multisets are the ones of Example 3. Let $P_4 = (K_4 = \{\varphi_{11}, \ldots, \varphi_{16}\}, \succeq_4 = K_4 \times K_4)$. Let us consider the context C_1 of Example 3, the respective distances of φ_{11}, φ_{12}, φ_{13}, φ_{14} and φ_{15} to C_1 are: 0, 2, 1, 1 and 4 (see Figure 1). Note that $d(\varphi_{16}, C_1)$ is not defined since the intersection of symbols is empty. However, $\lambda_{\text{syn}}(P_4, C_1) = (K_{4C_1}, \succeq_{4C_1})$ with $K_{4C_1} = K_4$: all the formulas are selected since the graph is connected. $\succeq_{4C_1}$ is s.t. $\varphi_{11} \succ_{4C_1} \varphi_{13} \simeq_{4C_1} \varphi_{14} \succ_{4C_1} \varphi_{12} \succ_{4C_1} \varphi_{15} \succ_{4C_1} \varphi_{16}$.

Example 4. *We consider a knowledge base K_5 which results from the aggregation of another famous example (Nixon Diamond) to the four formulas of Example 1:*

φ_5.	$r \to \neg pa$	*Republicans are not pacifists*
φ_6.	$q \to pa$	*Quakers are pacifists*
φ_7.	$q \to a$	*Quakers are Americans*
φ_8.	$a \to bb$	*Americans love baseball*
φ_9.	$q \to \neg bb$	*Quakers do not love baseball*
φ_{10}.	$q \wedge r$	*Nixon is a Quaker and republican*

Let $P_5 = (K_5 = \{\varphi_1, \ldots \varphi_{10}\}, \succeq_5 = \{(\varphi_x, \varphi_y)$ s.t. $x \le y\})$ with $C_7 = (q, r)$, then the filtered base with λ_{syn} is $K_{5C_7} = \{\varphi_5, \varphi_6, \varphi_7, \varphi_8, \varphi_9, \varphi_{10}\}$ with the ordering $\varphi_{10} \succ_{5C_7} \varphi_5 \succ_{5C_7} \varphi_6 \succ_{5C_7} \varphi_7 \succ_{5C_7} \varphi_9 \succ_{5C_7} \varphi_8$ because the distance paths from (q, r) to φ_5, φ_6, φ_7,φ_8,φ_9,φ_{10}, are respectively (2), (2), (2), (2,2), (2), (0). Then among the equivalently distant formulas the initial order applies.

3.3 Context-based desirable axioms

In Section 3.1, we have shown that when the context is fixed contextual inference is a rational non-monotonic inference, in this section we study the impact of changing the context on the inference relation. One benefit of taking into account the syntax relevance is the ability to impose that the context and the inferred conclusion are related as shown in the following example.

Example 5. *Let $P_6 = (K_6 = \{a, a \to b\}, \succeq_6 = K_6 \times K_6)$ with a context c (a fresh symbol), it holds that $\not\mid\!\sim^{\lambda_0}_{P_6,(c)} b$ and $\not\mid\!\sim^{\text{syn}}_{P_6,(c)} b$ indeed there is no syntax relevance between the context c and b. It would not be the case with lexicographic inference based on P_6, namely, it holds that $\mid\!\sim^{\text{lex}}_{P_6} b$. Even in classical logic, we would have $P_6 \cup \{c\} \models b$ which might be deemed irrelevant (à la relevance logic [1], but in a weaker form).*

Moreover, syntactic inference imposes that the pieces of knowledge of K that are used should be relevant with the context even indirectly as shown with $P_7 = (\{a, a \to b, d \to a, d \to c\}, \succeq_7 = K_7 \times K_7)$, we have $\mid\!\sim^{\text{syn}}_{P_7,(c)} b$ which is conform to $\mid\!\sim^{\text{lex}}_{P_7} b$.

Concerning classical logic, the previous example highlights that in the case of a consistent knowledge base, the contextual inference based on the refiner λ_{syn} recovers the results of classical inference for any query whose symbols are in the context.

Proposition 3 (Classical logic recovery)**.** *Given a consistent knowledge base $K \subseteq \mathcal{L}$, a formula $\varphi \in \mathcal{L}$ and a context $C \in \mathcal{MS}$ such that $\text{ms}(\varphi) \subseteq C$,*

$$K \models \varphi \quad \textit{iff} \quad \mid\!\sim^{\text{syn}}_{(K, K \times K), C} \varphi.$$

Proof. Since Lexicographic inference from a stratified consistent knowledge base K amounts to take all the formulas from each stratum (thus to ignore the stratification) and reason with classical inference from K, then it is enough to reason about the classical inferences from the compartment of K selected in the context C, namely $K_C = \{\psi \in K$ s.t. ψ is s-connected to C in $U_K\}$.

($\Rightarrow$) Let us assume (1) that $K \models \varphi$ but (2) that $\not\mid\!\sim^{\text{syn}}_{(K, K \times K), C} \varphi$, i.e., $K_C \not\models \varphi$. Note that $\varphi \not\equiv \bot$ since K is consistent and (1), note also that $\varphi \not\equiv \top$ because of (2). Now due to the deduction theorem of propositional logic, (1) translates into $Cl(K) \cup Cl(\{\neg\varphi\}) \vdash \bot$ and (2) into $Cl(K_C) \cup Cl(\{\neg\varphi\}) \nvdash \bot$ where $Cl(E)$ is the set of clauses (disjunction of literals) equivalent to the set of propositional formulas E and "$\vdash\bot$" means that the empty clause can be obtained by resolution. Now the fact that (1) holds together with (2) means that there is a subset $A \subseteq K$ with $A \cap K_C = \emptyset$ such that $Cl(K_C) \cup Cl(A) \cup Cl(\{\neg\varphi\}) \vdash \bot$. In order to obtain the empty clause by resolution, A should contain at least a formula with a common symbol either with a clause of $Cl(K_C)$ or with a clause of $Cl(\{\neg\varphi\})$, in both cases this formula is s-connected with C. Hence this formula should have been in K_C: contradiction.

($\Leftarrow$) Assume now that $K_C \models \varphi$ it means that $K \models \varphi$ since classical inference is monotonic. □

We are now going to state conditions on the relation between the context and the conclusion in order to recover plain lexicographic entailment. The next proposition follows from the result established in [8] stating that lexicographic inference satisfies syntax splitting and in particular the (Rel) axiom. The (Rel) axiom defined in [9] states that if the knowledge base can be divided in two compartments with different vocabularies and α and β are on the same compartment then the inference from α to β is equivalent whether or not the other compartment of the knowledge base is taken into account. In our approach, this corresponds to the case where the knowledge base is split into a compartment of formulas s-connected to the context C and another regrouping the remainder of the knowledge base.

Proposition 4. *Under the conditions that C s-connected to β and α s-connected to β in U_K, it holds that $\alpha \mid\!\sim^{\text{syn}}_{P,C} \beta$ iff $\alpha \mid\!\sim^{\text{lex}}_{P} \beta$.*

Proof. The proof is similar to the one of [8], by considering formulas that are connected to C as belonging to the same sub-language. Using

(Rel), α and β are both s-connected to C, hence the entailment only needs formulas s-connected to C: $\alpha \mid\!\sim^{\text{lex}}_{P} \beta$ iff $\alpha \mid\!\sim^{\text{lex}}_{\lambda(P,C)} \beta$. Due to the fact that $\mid\!\sim^{\text{syn}}_{P,C} = \mid\!\sim^{\text{lex}}_{\lambda(P,C)}$, we get the result. □

In the following, we define three axioms relative to the context. The first definition imposes a syntactic relevance between the context and the conclusion. Note that in [9] no context was taken into account to compute syntactic relevance. However, as seen in Prop. 4's proof, considering the s-connected component associated with the context, allows us to compartmentalize the knowledge base into two parts. With this in mind, our definition of context relevance is different from the (Rel) axiom of [9] which is more related with our two other axioms.

Definition 11 (Context relevance). *A contextual inference relation $\mid\!\sim^{\lambda}_{P}$ based on a prioritized knowledge base $P = (K, \succeq)$ and a refiner λ is* context relevant *iff for any context $C \in \mathcal{MS}$, and any consistent formulas $\alpha, \beta \in \mathcal{L}$ s.t. $\alpha \not\models \beta$*

if $\alpha \mid\!\sim^{\lambda}_{P,C} \beta$ then C is s-connected to β in U_K.

Here is a variant of (Ind) defined in [9] where independence was used in order to extend a premise while here it concerns a context to extend. More precisely, it expresses that when there is no syntactic path from a conclusion β to a formula, then this conclusion is also obtained from the same premise in a context extended with the symbols of this formula.

Definition 12 (Context independence). *A contextual inference relation $\mid\!\sim^{\lambda}_{P}$ based on a prioritized knowledge base $P = (K, \succeq)$ and a refiner λ satisfies* context independence *iff for any context $C \in \mathcal{MS}$ and any formulas $\alpha, \beta, \varphi \in \mathcal{L}$ such that α is connected to β in U_K, if φ is not s-connected to β in U_K and $\alpha \mid\!\sim^{\lambda}_{P,C} \beta$ then $\alpha \mid\!\sim^{\lambda}_{P,C \sqcup \text{ms}(\varphi)} \beta$.*

Next definition considers that if the syntactic similarity induced from C is the same that the one induced from C', then inference based on these two contexts should be the same.

Definition 13 (Context equivalence). *A contextual inference relation $\mid\!\sim^{\lambda}_{P}$ based on a prioritized knowledge base $P = (K, \succeq)$ and a refiner λ satisfies* context equivalence *iff for any contexts $C, C' \in \mathcal{MS}$ such that $\forall \varphi, \varphi' \in K$, $\varphi >^{\text{syn}}_{U_K,C} \varphi'$ iff $\varphi >^{\text{syn}}_{U_K,C'} \varphi'$, then $\alpha \mid\!\sim^{\lambda}_{P,C} \beta$ iff $\alpha \mid\!\sim^{\lambda}_{P,C'} \beta$*

The following proposition shows that the syntactic entailment $\mid\!\sim^{\text{syn}}$ is well behaved wrt our three axioms.

Proposition 5. *Given a prioritized knowledge base P, $\mid\!\sim^{\text{syn}}_{P}$ satisfies* context relevance*,* context independence *and* context equivalence*.*

Proof. (context relevance) Let $\alpha, \beta \in \mathcal{L}$ be consistent formulas s.t. $\alpha \not\models \beta$ and $C \in \mathcal{MS}$, assume that $\alpha \mid\!\sim^{\text{syn}}_{P,C} \beta$ then it means that every Lex-preferred α-consistent subbase B of $\lambda(P, C)$ is s.t. (1) $B \cup \{\alpha\} \models \beta$ where $P_C = (K_C, \succ_C)$ and K_C contains only formulas s-connected to C in U_K. (1) is equivalent to $B \cup \{\alpha\} \cup \{\neg\beta\}$ is inconsistent. Since $\alpha \not\models \beta$ it means that $\{\alpha\} \cup \{\neg\beta\}$ is consistent, hence B should contain at least one formula that shares some variable with β, it means that β is s-connected to a formula in B, since all formulas in B are s-connected with C.

(context independence) The proof is based on Proposition 4 by considering the knowledge base $K_{C \sqcup \text{ms}(\varphi)}$ which can be split into K_C and the rest since φ is not s-connected to β.

(context equivalence) If C and C' induce the same ranking on the formulas of K, then the Lex-preferred α-consistent subbases based on $\succeq_C$ and $\succeq_{C'}$ are the same. □

3.4 Optional syntax-based axioms

Due to the fact that the context is a multiset of symbols, refiners are naturally designed to take into account the syntax of the context. Hence, assuming that the context can be represented by a formula, different semantically equivalent ways to write this formula may influence the result since each can have a different associated multiset.

Definition 14 (Sensitivity to context syntax). *A contextual inference relation $\mid\!\sim^{\lambda}_{P}$ based on a prioritized knowledge base $P = (K, \succeq)$ and a refiner λ is* sensitive to context syntax *iff there exist two contexts expressed by the formulas φ, φ' and two formulas $\alpha, \beta \in \mathcal{L}$ such that $\varphi \equiv \varphi'$ and $\alpha \mid\!\sim^{\lambda}_{P,\text{ms}(\varphi)} \beta$ but $\alpha \not\mid\!\sim^{\lambda}_{P,\text{ms}(\varphi')} \beta$.*

Proposition 6. *λ_0 and λ_{syn} are* sensitive to context syntax.

Proof. See Example 2 (continued). □

Example 2 (continued): *Let us consider P_2 of Example 2, we have seen that the context $\varphi = b$ leads to $\lambda_0(P_2, (b)) = (\{b, a \to \neg b\}, K_{2(b)} \times K_{2(b)})$ so $\top \mid\!\sim^{\lambda_0}_{P,(b)} b$. Now considering the context corresponding to $\varphi' = b \wedge (c \vee \neg c)$ whose multiset is $\text{ms}(\varphi') = (b, c, c)$, $\lambda_0(P_2, (b, c, c)) = K_2, K_2 \times K_2)$ hence $\top \not\mid\!\sim^{\lambda_0}_{P,(b,c,c)} b$. The following table shows the distance paths from the contexts (b) and (b, c, c) to each formula of K_2.*

	C	
K_2	(b)	(b, c, c)
b	(0)	(2)
c	$(1, 2, 1)$	(2)
$c \to a$	$(1, 2)$	(3)
$a \to \neg b$	(1)	(3)

Hence $\lambda_{\text{syn}}(P_2, (b)) = (\{b\}, \{a \to \neg b\}, \{c \to a\}, \{c\})$ and $\lambda_{\text{syn}}(P_2, (b, c, c)) = (\{b, c\}, \{a \to \neg b, c \to a\})$ which results in $\top \mid\!\sim^{\text{syn}}_{P_2,(b)} b \wedge \neg c$ and $\top \mid\!\sim^{\text{syn}}_{P_2,(b,c,c)} b \wedge c$. In both cases $\varphi \equiv \varphi'$ but entailment in the context of their respective multisets $\text{ms}(\varphi)$ and $\text{ms}(\varphi')$ leads to different conclusions.

Obviously the void refiner λ_{void} (such that for any prioritized knowledge base and for any context $\lambda_{void}(P, C) = P$) is not sensitive to context syntax. Recall that a void refiner is equivalent to the plain lexicographic entailment. Another interesting subject is to check whether the way each formula of the knowledge base is written may influence inference.

Definition 15 (Sensitivity to knowledge syntax). *A contextual inference relation $\mid\!\sim^{\lambda}_{P}$ based on a prioritized knowledge base $P = (K, \succeq$) and a refiner λ is* sensitive to knowledge syntax *iff there exist a context C and four formulas $\varphi, \varphi', \alpha, \beta \in \mathcal{L}$ such that $\varphi \in K$, $\varphi \equiv \varphi'$ and $\alpha \mid\!\sim^{\lambda}_{P,C} \beta$ but $\alpha \not\mid\!\sim^{\lambda}_{P',C} \beta$ where P' is P in which φ' replaces φ.*

Proposition 7. *λ_0 and λ_{syn} are sensitive to knowledge syntax.*

Proof. See Example 6. □

Example 6. *Let $P_8 = (K_8 = \{a \to b, a \wedge c \to \neg b\}, \succeq_8 = K_8 \times K_8)$ and let us define P'_8 as the variant of P_8 where $a \to b$ is replaced with the equivalent formula $a \wedge (c \vee \neg c) \to b$, then $\lambda_0(P_8, (c)) = (\{a \wedge c \to \neg b\})$ and $\lambda_0(P'_8, (c)) = (\{a \wedge (c \vee \neg c) \to b, a \wedge c \to \neg b\})$. Thus, $a \wedge c \mid\!\sim^{\lambda_0}_{P_8,(c)} \neg b$ but $a \wedge c \not\mid\!\sim^{\lambda_0}_{P'_8,(c)} \neg b$. Let us now define P''_8 as the variant of P_8 where $a \to b$ is replaced with the equivalent formula $a \wedge a \to b$. The path distances from (a) to the formulas*

$a \to b$, $a \wedge c \to \neg b$, $a \wedge a \to b$ are respectively (1),(2),(2) leading to $\lambda_{\text{syn}}(P_8, (a)) = (\{a \to b\}, \{a \wedge c \to \neg b\})$ and $\lambda_{\text{syn}}(P_8'', (a)) = P_8''$ which means that $a \mid\!\sim^{\text{syn}}_{P_8,(a)} b$ while $a \mid\!\not\sim^{\text{syn}}_{P_8'',(a)} b$.

3.5 Complexity discussion

Let us end this section with a comment about complexity. Unfortunately, there is no theoretical worse-case complexity gain with the contextual inference approach because in the worst case (when the knowledge base is consistent and contains only formulas relevant to the query) the refiner will return the entire initial knowledge base. However, in the particular case where the knowledge base contains several distinct parts, a context could concern only some of these parts. In this case, by definition, contextual inference will be less (or equally) costly than a classical lexicographic inference since computations would involve a compartment of formulas smaller or equal to the whole knowledge base.

Concerning the computation of the compartmentalization, it is not negligible but polynomial, as shown below.

Proposition 8. *Given a knowledge base $K = \{\varphi_1, \ldots, \varphi_n\}$ with formulas of maximum size k ($\forall i \in [1, n], |\operatorname{ms}(\varphi_i)| \leq k$) where $k \ll n$, and a context $C \in \mathcal{MS}$ s.t. $|C| \ll n$, the complexity of the extraction with λ_0 is linear in n while the extraction with λ_{syn} is polynomial in n.*

Proof. Both operators need to work on the multisets of symbols associated to the formulas of the knowledge base, building these multisets is done in linear time wrt k and can be done once and for all.
• λ_0 operator only requires to perform n multisets intersection between the formulas and the context, each intersection being linear in $\max(k, |C|)$. Hence the worse case complexity of compartmentalizing K with the refiner λ_0 is in $\Theta(n)$.
• λ_{syn} operators require to compute n distances: from C to each formula of the knowledge base. Theses distances are comparisons of multisets of maximum size $\max(k, |C|)$, hence the distance computation is in linear time with $\max(k, |C|) \ll n$. Next, λ_{syn} requires to find the shortest paths from the context C to each formula of K (e.g. using Dijkstra's algorithm, computing one shortest path requires $n \log n$ operations). This leads to a complexity of the compartmentalization with λ_{syn} in $\Theta(n^2 \log n)$. □

While arguably reasonable, compartmentalization still has a cost, it should be noted that it can be computed once and for all in the case where the context remains the same during several queries, in which case we could benefit from low amortized complexity.

4 Conclusion

The present paper deals with how to assess arguments and answer queries given an inconsistent or not, prioritized or not, knowledge base, using a kind of compartmentalization. This compartmentalization is based on the relevance with a context, which is performed by a selection function called refiner. This proposal is a first step in the general goal of simulating a rational agent and trying to explain how this agent can exploit a knowledge base built incrementally with no moderation. This paper aims at introducing a new notion, which we think was not exploited enough in the literature, namely to take into account the way formulas are written in order to select more efficiently the accurate pieces of knowledge for answering queries. Indeed, in order to build systems that can help human to reason, it is important to take into account the form, in addition to the meaning. This "syntax-dependent by design" approach, where e.g. $(a \wedge a)$ can be managed differently from a if wanted, is a proposal in that way.

One first benefit of the proposed approach is to determine, on the fly, specific preferences on the formulas of the knowledge base. Compared to other approaches that do not take preferences as input, contextual inference is able to make more decisions.

Coming back to the initial question: "what conditions must the refiner meet to ensure that the syntactic selection of the subbase allows for sound and complete reasoning", we stress that, when the knowledge base is consistent, there is no risk of over-sensitivity (Prop. 3). Indeed, the conclusion obtained on a filtered subbase will be the same as the one obtained by classical inference on the whole base.

In addition, our approach allows for a form of relevance in a propositional setting (presented in Section 3.3), by providing three desirable axioms to characterize relevant refiners. There is a huge amount of work about relevance logic [1] were the idea is to redefine classical logic in a way that relevance is obtained by design. Quoting [15]: "the *variable sharing* principle says that no formula of the form $A \to B$ can be proven in a relevance logic if A and B do not have at least one propositional variable [...] in common and that no inference can be shown valid if the premises and conclusion do not share at least one propositional variable." In our work, $A \to B$ translates into $A \mid\!\sim_{P,C} B$ (where P is a possibly empty knowledge base) and $\mid\!\sim$ is based on classical inference $\models$ from a subset of P according to C. An extension of this work would be to replace $\models$ with an inference operator from relevance logic.

A long term objective of the "syntax-dependent by design" approach is to be able to quantify the effect of using some words for rhetoric and persuasion, for instance by comparing the conclusions obtained with syntactic sensitivity to the one obtained without it. Similarly, a perspective would be to quantify the redundancies and repetitions in a discourse. The proposed approach would need to be extended since it does not yet take into account the connectors. Indeed, when a variable appears twice, connectors seem necessary to know if there is indeed a redundancy: namely $a \wedge (a \to b)$ is not as redundant as $(a \wedge a) \to b$. One way of quantifying redundancies would be to compare the formula with one of its canonical form (as in e.g. [11]).

Moreover, it is worth noticing that the idea to take into account the variable symbols in order to focus on the relevant pieces of information can be extended to not only check whether exactly the same variable symbol is present, but also to check whether another symbol that belongs to the same lexical field is present (given extra-knowledge about lexical fields or ontological relations between symbols, or chunks). This would be particularly useful when considering formulas directly translated from English sentences, as the notions behind the words are meaningful. For instance, words such as scent, aroma, fragrance and smell are all related to the concept of odor, but their connoted meanings are quite different and oftentimes specific to the person hearing them. The ability to handle connoted meanings of symbols, e.g. by integrating positive or negative emotions associated with them, is a promising perspective for which the syntax-sensitive inference operator could be adapted.

Acknowledgments

We deeply thank the reviewers, the paper greatly benefited from all the suggested improvements and corrections.

References

[1] A. R. Anderson, N. D. Belnap Jr, and J. M. Dunn. *Entailment, Vol. II: The logic of relevance and necessity*, volume 5027. Princeton University Press, 2017.

[2] J. R. Anderson, D. Bothell, M. D. Byrne, S. Douglass, C. Lebiere, and Y. Qin. An integrated theory of the mind. *Psychological review*, 111 (4):1036, 2004.

[3] S. Benferhat, C. Cayrol, D. Dubois, J. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *IJCAI*, volume 93, pages 640–645, 1993.

[4] S. Benferhat, D. Dubois, and H. Prade. Nonmonotonic reasoning, conditional objects and possibility theory. *Artificial Intelligence*, 92(1-2): 259–276, 1997.

[5] G. Brewka. Preferred subtheories: An extended logical framework for default reasoning. In *IJCAI*, volume 89, pages 1043–1048. Citeseer, 1989.

[6] M. Goldszmidt and J. Pearl. On the consistency of defeasible databases. *Artificial Intelligence*, 52(2):121–149, 1991.

[7] P. Gärdenfors and D. Makinson. Nonmonotonic inference based on expectations. *Artificial Intelligence*, 65(2):197–245, 1994. ISSN 0004-3702. doi: https://doi.org/10.1016/0004-3702(94)90017-5. URL https://www.sciencedirect.com/science/article/pii/0004370294900175.

[8] J. Heyninck, G. Kern-Isberner, and T. Meyer. Lexicographic entailment, syntax splitting and the drowning problem. In *31st International Joint Conference on Artificial Intelligence, IJCAI*, 2022.

[9] G. Kern-Isberner, C. Beierle, and G. Brewka. Syntax splitting= relevance+ independence: New postulates for nonmonotonic reasoning from conditional belief bases. In *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning*, volume 17, pages 560–571, 2020.

[10] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial intelligence*, 44 (1-2):167–207, 1990.

[11] J. Lang, P. Liberatore, and P. Marquis. Propositional independence: Formula-variable independence and forgetting. *J. Artif. Intell. Res.*, 18: 391–443, 2003. doi: 10.1613/jair.1113. URL https://doi.org/10.1613/jair.1113.

[12] D. Lehmann. Another perspective on default reasoning. *Annals of mathematics and artificial intelligence*, 15:61–82, 1995.

[13] M. H. Liffiton and K. A. Sakallah. Algorithms for computing minimal unsatisfiable subsets of constraints. *Journal of Automated Reasoning*, 40(1):1–33, 2008.

[14] D. Makinson. General theory of cumulative inference. In *Non-Monotonic Reasoning: 2nd International Workshop Grassau, FRG, June 13–15, 1988 Proceedings 2*, pages 1–18. Springer, 1989.

[15] E. Mares. Relevance logic. *The Stanford Encyclopedia of Philosophy (Fall 2022 Edition)*, online: https://plato.stanford.edu/archives/fall2022/entries/logic-relevance/, 2022.

[16] B. Nebel. Belief revision and default reasoning: Syntax-based approaches. *KR*, 91:417–428, 1991.

[17] J. Pearl. System Z: A natural ordering of defaults with tractable applications to nonmonotonic reasoning. In *Proceedings of the 3rd Conference on Theoretical Aspects of Reasoning about Knowledge*, pages 121–135, 1990.

[18] G. Pinkas and R. P. Loui. Reasoning from inconsistency: A taxonomy of principles for resolving conflict. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, pages 709–719, 1992.

[19] N. Rescher and R. Manor. On inferences from inconsistent premises. *Theory and Decision*, 1(2):179–217, 1970. doi: 10.1007/bf00154005.

[20] Y. Shoham. Nonmonotonic logics: Meaning and utility. In *IJCAI*, volume 10, pages 388–393. Citeseer, 1987.