Similarity Measures for First-Order Logical Arguments

Victor David[1], Jérôme Delobelle[2] and Jean-Guy Mailly[3,*]

[1]University Côte d'Azur, Inria, CNRS, I3S, Sophia-Antipolis, France
[2]Université Paris Cité, LIPADE, Paris, France
[3]Université Toulouse Capitole, IRIT, Toulouse, France

Abstract

Similarity in formal argumentation has recently gained attention due to its significance in problems such as argument aggregation in semantics and enthymeme decoding. While prior work has focused on propositional logic arguments, we extend these approaches to First-Order Logic (FOL) arguments, enabling reasoning based on the similarity of arguments in more complex and realistic contexts. We present a comprehensive framework for FOL argument similarity, including: 1. An extended axiomatic foundation for similarity measures, 2. A parametric model decomposed into four levels to efficiently evaluate structured knowledge, 3. A Tversky-based family of measures to instantiate these concepts, 4. A set of constraints ensuring well-behaved models that satisfy axioms, and 5. We introduce and analyze non-symmetric similarity measures in formal argumentation for the first time.

Keywords

Similarity Measure, First-Order Logic, Argumentation

1. Introduction

An argumentation system typically consists of two main components: a *representation component*, which structures information as a graph with arguments as nodes and binary relations between them (either positive, called *support*, or negative, called *attack*), and a *reasoning component*, which determines the acceptability of arguments according to argumentation semantics.

Formal argumentation has become an important area within knowledge representation and reasoning, with applications in domains such as decision-making [1], explainable artificial intelligence (XAI) [2, 3], judgmental forecasting [4, 5], and enthymeme-based reasoning [6, 7, 8].

Across these diverse applications, a common concern has emerged: the need to reason not only about the presence of arguments and relations, but also about the degree of *similarity* between arguments. This notion, initially introduced in [9] for propositional arguments, has been formalized within an axiomatic framework to ensure that similarity measures satisfy desirable rational properties. Subsequent work [10, 11] has shown the practical benefits of these measures, in particular for avoiding redundancy and improving argument aggregation in *gradual semantics*, where each argument receives a numerical degree of acceptability. An argument is defined as a pair $\langle \text{Premises}, \text{Claim} \rangle$, where the premises aim to justify the claim. For instance, consider two arguments $A_1 = \langle \{p_1, p_1 \to q\}, q \rangle$ and $A_2 = \langle \{p_2, p_2 \to q\}, q \rangle$, which support the same claim q through different reasoning. These arguments can be considered having some similarity, as they both justify the same claim. Now suppose they both attack a third argument $B = \langle \{\neg q \wedge \neg r\}, \neg q \wedge \neg r \rangle$. In gradual semantics, the degree of acceptability of an argument is often influenced by the number and the degree of acceptability of its attackers. Assuming that A_1, A_2, and $A_3 = \langle \{p_3, p_3 \to r\}, r \rangle$ all have equal degree of acceptability, the combined impact of the "similar" arguments A_1 and A_2 on B should be lower than the combined impact of two "dissimilar" arguments, such as A_1 and A_3. The similarity between A_1 and A_2 reflects a form of redundancy, which should reduce their aggregated influence when computing the acceptability of B. In this setting, similarity scores between arguments provide a principled way to adjust aggregation in order to avoid overestimating the effect of repeated or equivalent information.

NMR'25: 23rd International Workshop on Nonmonotonic Reasoning, November 11-13, 2025, Melbourne, Australia
*Corresponding author.
victor.david@inria.fr (V. David); jerome.delobelle@u-paris.fr (J. Delobelle); jean-guy.mailly@irit.fr (J. Mailly)
0000-0002-4216-0876 (V. David); 0000-0003-1691-4731 (J. Delobelle); 0000-0001-9102-7329 (J. Mailly)

Furthermore, the role of similarity has recently been extended beyond argument aggregation. In the context of enthymeme decoding, similarity serves as one of the evaluation criteria to assess the quality of a decoding (i.e. a reconstructed argument). More specifically, it quantifies how faithful a decoding is to the original enthymeme (i.e. an incomplete argument) [7]. In this setting, similarity helps ensure that the decoding preserves the informational content of the original.

Overall, similarity between arguments should be seen as a core structural feature of argumentation systems, just like attack and support relations. However, existing studies on logical argument similarity remain largely confined to the propositional level, which limits their applicability. Many real-world arguments involve quantifiers, relational structures, and variables that propositional logic cannot adequately express. In this work, we address this gap by introducing a general framework for evaluating similarity between arguments expressed in first-order logic (FOL). We extend existing principles analysis to the FOL setting, and propose a four-level parametric model that captures the internal structure of FOL arguments, from atomic literals to sets of formulae. The framework is instantiated using Tversky-based similarity functions, guided by formal constraints that ensure compliance with the principles. Additionally, we provide an analysis of non-symmetric similarity in formal argumentation, highlighting its potential role in capturing directional similarities between arguments.

This framework is illustrated through examples but remains application-independent, making it suitable for integration into various reasoning settings, including but not limited to gradual semantics and enthymeme reconstruction. By moving beyond propositional logic, we aim to provide a more expressive and realistic foundation for reasoning about argument similarity in complex domains.

This paper simplifies our previous approach based on Order-Sorted (OS) FOL [12] by adopting standard, unsorted FOL. OS-FOL is more expressive in theory, but it requires explicit sort declarations and type-consistent instantiations, which are impractical in real-world settings, especially for natural language arguments, where such type information is often implicit or unavailable. By removing these constraints, the new framework becomes easier to apply and enables a reformulation of the principles based directly on similarities between predicates and constants, resulting in more precise definitions. Moreover, we revise the definition of instantiated arguments to ensure valid arguments (e.g. when instantiating existential claims, only those instances actually supported by the premises are retained). We also provide a deeper analysis of both symmetric and non-symmetric similarity models, including a characterization of the conditions under which total similarity (i.e., value 1) is achieved (Theorem 3). Finally, we include a brief related work section on logical similarity measures to situate our contribution.

2. Logic and Arguments

First Order Logic is a rich framework that develops information about the objects and can also express the relationships between them (using predicates). For instance, let the constants $t = Tropical\_Forest$, $u = Uganda\_Forest$, and the predicates $F(x) = Forest(x)$, $D(x) = massive\_Deforestation(x)$, and $L(x) = Lose\_biodiversity(x)$. Using these, we can form statements like, $F(t)$: "The tropical forest is a forest" or $\forall x F(x) \wedge D(x) \rightarrow L(x)$: "Any forest that undergoes massive deforestation will lose biodiversity".

Definition 1. A **First Order Language** FOL, is a set of formulae built up by induction from: a set $\mathbb{C}$ of constants ($\mathbb{C} = \{a_1, \ldots, a_l\}$), a set $\mathbb{V}$ of variables ($\mathbb{V} = \{x, y, z, \ldots\}$), a set $\mathbb{P}$ of predicates ($\mathbb{P} = \{P_1, \ldots, P_m\}$), a function $\mathtt{ar} : \mathbb{P} \rightarrow \mathbb{N}$ which gives the arity of predicates, the usual connectives ($\neg$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$), Boolean constants $\top$ (true) and $\bot$ (false) and quantifier symbols ($\forall, \exists$). A *grounded formula* is a formula without any variable.

We use lowercase greek letters (*e.g.* ϕ, ψ) to denote formulae, and uppercase ones (*e.g.* Φ, Ψ) to denote sets of formulae. The set of all FOL formulae is denoted by FOL. We assume formulae to be *prenex*, *i.e.* written as $\mathcal{Q}_1 x_1, \ldots, \mathcal{Q}_k x_k \phi$ where $\mathcal{Q}_j$ is a quantifier (for each $j \in \{1, \ldots, k\}$) and ϕ is a non-quantified formula. A formula ϕ is in negation normal form (NNF) iff it does not contain implication or equivalence symbols, and every negation symbol occurs directly in front of an atom (*i.e.,* a predicate with its parameters). Following [13], we slightly abuse words and denote by $\texttt{NNF}(\phi)$ the formula in NNF obtained from ϕ by "pushing down" every occurrence of $\neg$ (using De Morgan's law) and eliminating double negations. For instance, $\texttt{NNF}(\neg((P(a) \rightarrow Q(a)) \vee \neg Q(b))) = P(a) \wedge \neg Q(a) \wedge Q(b)$. In that case, we call *literal* either an atom or the negation of an atom. The set of grounded atoms is denoted by $\mathbb{A}$. We denote by $\texttt{Lit}(\phi)$ the set of literals occurring in $\texttt{NNF}(\phi)$, hence $\texttt{Lit}(\neg((P(a) \rightarrow Q(a)) \vee \neg Q(b))) = \{P(a), \neg Q(a), Q(b)\}$. For a given set of predicates $\mathbb{P}$, we define $\mathbb{L} = \{P(x_1, \ldots, x_k), \neg P(x_1, \ldots, x_k) \mid P \in \mathbb{P}, \texttt{ar}(P) = k, k \geq 0\}$ the set of literals. We say that a literal L is *negative* when it starts with a negation, denoted by $\texttt{Pol}(L) = -$. Otherwise it is *positive* and is denoted by $\texttt{Pol}(L) = +$. Two literals L and L' have the same *polarity* if $\texttt{Pol}(L) = \texttt{Pol}(L')$. Finally, given a grounded literal $L = \pm P(a_1, \ldots, a_k)$ where $\pm \in \{+, -\}$ indicates the polarity of L, $\texttt{Pred}(L) = P$ is the name of the predicate underlying L, and $\texttt{Para}(L) = \langle a_1, \ldots, a_k \rangle$. Consider $\phi \in \texttt{FOL}$, ϕ is in conjunctive normal form (CNF) if it is a conjunction of clauses $\bigwedge_i \delta_i$ where each clause δ_i is a disjunction of literals $\bigvee_j l_j$. For instance $P(a) \wedge (Q(a) \vee Q(b))$ is in CNF while $(P(a) \wedge Q(a)) \vee Q(b)$ is not. Clauses can be represented as sets of literals, and CNF formulae as sets of clauses.

$\texttt{FOL}$ formulae are evaluated via a notion of structure, *i.e.* a triplet $\mathbf{St} = (D, Rel, Cons)$ where D is the (non-empty) domain, $Rel = \{R_1, \ldots, R_m\}$ are relations over the domain, and $Cons = \{c_1, \ldots, c_l\}$ are constants in the domain.

Definition 2. An **interpretation** $\mathbf{I_{St}}$ over a structure $\mathbf{St}$ assigns to elements of the $\texttt{FOL}$ vocabulary some values in the structure $\mathbf{St}$. Formally,
- $\mathbf{I_{St}}(P_j) = R_j$, for $j \in \{1, \ldots, m\}$ (each predicate symbol is assigned to a relation),
- $\mathbf{I_{St}}(a_j) = c_j$, for $j \in \{1, \ldots, l\}$ (each constant symbol is assigned to a constant value).

Then satisfaction of formulae is recursively defined by:
- $\mathbf{I_{St}} \models P_j(x_1, \ldots, x_k)$ iff $(x_1, \ldots, x_k) \in R_j$,
- $\mathbf{I_{St}} \models \exists x \phi$ iff $\mathbf{I}_{\mathbf{St}, x \leftarrow v} \models \phi$ for some $v \in D$,
- $\mathbf{I_{St}} \models \forall x \phi$ iff $\mathbf{I}_{\mathbf{St}, x \leftarrow v} \models \phi$ for each $v \in D$,
- conjunctions, disjunctions and negations are interpreted as usually in classical logic,

where $\mathbf{I}_{\mathbf{St}, x \leftarrow v}$ is a modified version of $\mathbf{I_{St}}$ s.t. the variable x is replaced by a value v in the domain D. Finally, if Φ is a set of formulae, then $\mathbf{I_{St}} \models \Phi$ iff $\mathbf{I_{St}} \models \phi$ for each $\phi \in \Phi$.

We say that Φ is *consistent* if there is at least one interpretation $\mathbf{I_{St}}$ s.t. $\mathbf{I_{St}} \models \Phi$.

We now define instantiations as grounded formulae compatible with a given $\texttt{FOL}$ formula.

Definition 3. Given Φ a set of $\texttt{FOL}$ formulae and $\mathbf{I_{St}}$ an interpretation over a structure $\mathbf{St}$, the set of **instantiations** of Φ is defined recursively by:
- $\texttt{Inst}_{\mathbf{I_{St}}}(\Phi) = \{\Phi\}$ if $\Phi = \{\phi\}$, where ϕ is a grounded formula s.t. $\mathbf{I_{St}} \models \phi$,
- $\texttt{Inst}_{\mathbf{I_{St}}}(\Phi) = \{\texttt{Inst}_{\mathbf{I_{St}}}(\{\phi_{x \leftarrow v} \mid \mathbf{I_{St}} \models \phi_{x \leftarrow v}, v \in D\})\}$ if $\Phi = \{\forall x \phi\}$,
- $\texttt{Inst}_{\mathbf{I_{St}}}(\Phi) = \{\texttt{Inst}_{\mathbf{I_{St}}}(\{\phi_{x \leftarrow v} \mid \mathbf{I_{St}} \models \phi_{x \leftarrow v}, v \in V\}) \mid \emptyset \subset V \subseteq D\}$ if $\Phi = \{\exists x \phi\}$,
- $\texttt{Inst}_{\mathbf{I_{St}}}(\Phi) = \{I_1 \cup I_2 \mid I_1 \in \texttt{Inst}_{\mathbf{I_{St}}}(\{\phi_1\}), I_2 \in \texttt{Inst}_{\mathbf{I_{St}}}(\Phi_2), \mathbf{I_{St}} \models I_1 \cup I_2\}$ if $\Phi = \{\phi_1\} \cup \Phi_2$ with $\phi_1 \notin \Phi_2$,

where $\phi_{x \leftarrow v}$ is the formula ϕ s.t. all the occurrences of the variable x are replaced by the value v.

The idea is that formulae with quantified variables may be instanciated in various ways. Assuming that for a predicate P and an interpretation $\mathbf{I_{St}}$, we have $\mathbf{I_{St}} \models \{P(a), P(b)\}$; then $\texttt{Inst}_{\mathbf{I_{St}}}(\exists x P(x)) = \{\{P(a)\}, \{P(b)\}, \{P(a), P(b)\}\}$ and $\texttt{Inst}_{\mathbf{I_{St}}}(\forall x P(x)) = \{\{P(a), P(b)\}\}$.

Moreover, an instantiation is consistent because of the constraint $\mathbf{I_{St}} \models I_1 \cup I_2$ in the last part of the definition. This constraint means that, if *e.g.,* we consider the set of formulae $\{\exists x P(x), \exists x \neg P(x)\}$, then we keep only the instantiations where $P(A)$ is true and $P(B)$ is false, or the opposite.

Example 1. Let $\Phi = \{\exists x\, F(x) \wedge D(x),\ \forall x\, F(x) \wedge D(x) \rightarrow L(x)\}$ be a set of formulae with the interpretation $\mathbf{I_{St}} \models \{F(t) \wedge D(t), F(u) \wedge D(u)\}$. We have $\texttt{Inst}_{\mathbf{I_{St}}}(\Phi) = \{$
$\{F(t) \wedge D(t), F(t) \wedge D(t) \rightarrow L(t), F(u) \wedge D(u) \rightarrow L(u)\}$,
$\{F(u) \wedge D(u), F(t) \wedge D(t) \rightarrow L(t), F(u) \wedge D(u) \rightarrow L(u)\}$,
$\{F(t) \wedge D(t), F(u) \wedge D(u), F(t) \wedge D(t) \rightarrow L(t), F(u) \wedge D(u) \rightarrow L(u)\}\}$.

From structure and interpretation, we define the consequence relation over FOL formulae.

Definition 4. Let ϕ and ψ be two FOL formulae. We say that ψ is a **consequence** of ϕ, denoted by $\phi \vdash \psi$, if for any structure $\mathbf{St}$, and any interpretation $\mathbf{I_{St}}$ over $\mathbf{St}$, $\mathbf{I_{St}} \models \phi$ implies $\mathbf{I_{St}} \models \psi$. Two formulae ϕ, ψ are *equivalent* (denoted $\phi \equiv \psi$) iff $\phi \vdash \psi$ and $\psi \vdash \phi$.

A *logic* is a pair $(L, \mathrel{|\!\sim})$ where L is a set of formulae (*i.e.* a language) and $\mathrel{|\!\sim} \subseteq L \times L$ is a consequence relation. An example of logic is $(\mathcal{L}, \vdash)$ with $\mathcal{L}$ an FOL language following Def. 1 and $\vdash$ the consequence relation from Def. 4. Logical arguments [14] are defined as follows:

Definition 5. An **argument** built under a logic $(L, \mathrel{|\!\sim})$ is a pair $\langle \Phi, \phi \rangle$, where[1] $\Phi \subseteq_f L$ and $\phi \in L$, s.t. Φ is consistent, $\Phi \mathrel{|\!\sim} \phi$, and $\nexists \Phi' \subset \Phi$ s.t. $\Phi' \mathrel{|\!\sim} \phi$. An argument $A = \langle \Phi, \phi \rangle$ is *trivial* iff $\Phi = \emptyset$ and $\phi \equiv \top$. Φ is called the support ($\mathtt{S}(A) = \Phi$) and ϕ its claim ($\mathtt{C}(A) = \phi$). The set of all arguments built under $(L, \mathrel{|\!\sim})$ is denoted $\mathtt{Arg}(L)$.

Example 2. Let $A, B \in \mathtt{Arg}(\mathcal{L})$ such that: $A = \langle \{\exists y\, F(y) \land D(y), \forall x\, F(x) \land D(x) \rightarrow L(x)\}, \exists y\, F(y) \land L(y) \rangle$, $B = \langle \{F(t), D(t), \forall x\, F(x) \land D(x) \rightarrow L(x)\}, F(t) \land L(t) \rangle$.

In this paper, we assume an FOL language $\mathcal{L}$ and focus on the arguments $\mathtt{Arg}(\mathcal{L})$ built under the logic $(\mathcal{L}, \vdash)$. Since logical arguments are mathematical objects satisfying some specific properties, instantiating them must be done with care, so we need to adapt the notion of instantiation of a set of formulae to take into account the particularities of arguments (for instance, the relation between the support and the claim or the minimality requirement).

Definition 6. Let $A = \langle \Phi, \phi \rangle$ be a logical argument and $\mathbf{I_{St}}$ an interpretation over a structure $\mathbf{St}$, the set of **instantiated arguments** of A is defined by:

$$\mathtt{Inst}_{\mathbf{I_{St}}}(A) = \{A^{\mathtt{i}} = \langle \Gamma, \bigwedge \Psi \rangle \mid \Gamma \subseteq \Delta \in \mathtt{Inst}_{\mathbf{I_{St}}}(\Phi), \Psi \in \mathtt{Inst}_{\mathbf{I_{St}}}(\{\phi\}), A^{\mathtt{i}} \in \mathtt{Arg}(\mathcal{L})\}$$

We add the power "$\mathtt{i}$" to an argument to indicate that we are considering its instantiated version.

Example 3. From Example 1 the instantiations of the support of A are: $\mathtt{Inst}_{\mathbf{I_{St}}}(\mathtt{S}(A)) = \{$
$\{F(t) \land D(t), F(t) \land D(t) \rightarrow L(t), F(u) \land D(u) \rightarrow L(u)\}$,
$\{F(u) \land D(u), F(t) \land D(t) \rightarrow L(t), F(u) \land D(u) \rightarrow L(u)\}$,
$\{F(t) \land D(t), F(u) \land D(u), F(t) \land D(t) \rightarrow L(t), F(u) \land D(u) \rightarrow L(u)\}\}$.
Then for the claim: $\mathtt{Inst}_{\mathbf{I_{St}}}(\mathtt{C}(A)) = \{\{F(t) \land L(t)\}, \{F(u) \land L(u)\}, \{F(t) \land L(t), F(u) \land L(u)\}\}$.
Hence $\mathtt{Inst}_{\mathbf{I_{St}}}(A) = \{A^{\mathtt{i}}_1 = \langle \{F(t) \land D(t), F(t) \land D(t) \rightarrow L(t)\}, F(t) \land L(t) \rangle$,
$A^{\mathtt{i}}_2 = \langle \{F(u) \land D(u), F(u) \land D(u) \rightarrow L(u)\}, F(u) \land L(u) \rangle, A^{\mathtt{i}}_3 =$
$\langle \{F(t) \land D(t), F(t) \land D(t) \rightarrow L(t), F(u) \land D(u), F(u) \land D(u) \rightarrow L(u)\}, F(t) \land L(t) \land F(u) \land L(u) \rangle\}$.
Similarly, $\mathtt{Inst}_{\mathbf{I_{St}}}(B) = \{B^{\mathtt{i}}_1 = \langle \{F(t), D(t), F(t) \land D(t) \rightarrow L(t)\}, F(t) \land L(t) \rangle\}$.

Two sets of formulae $\Phi, \Psi \subseteq_f \mathcal{L}$ are *equivalent*, i.e., $\Phi \cong \Psi$, iff there is a bijection $f : \Phi \rightarrow \Psi$ s.t. $\forall \phi \in \Phi$, $\phi \equiv f(\phi)$. We use this restricted equivalence notion to avoid equivalences that could be false due to incorrect information. For example the sets $\{Square(a), Square(a) \rightarrow Rectangle(a)\}$ and $\{Rectangle(a), Rectangle(a) \rightarrow Square(a)\}$ should not be equivalent. However, we may want to consider that a set of formulae is equivalent with the conjunction of its elements (*e.g.* $\{P(a), Q(a)\}$ and $\{P(a) \land Q(a)\}$ are equivalent). To make them equivalent, we borrow the method used in [15]. We transform every formula into a CNF, then we split it into a set containing its clauses. In our approach, we consider one CNF per formula. For that purpose, we use a finite sub-language $\mathcal{F}$ of $\mathcal{L}$ containing one formula per equivalent class and the formula should be in CNF. To ensure that $\mathcal{F}$ is finite, we assume that the logical signature is also finite, meaning that the number of predicate symbols, constants, and function symbols is bounded. This assumption is not too restrictive in practice. In real-world applications, such as natural language processing (where textual input could be translated into FOL representations), the vocabulary used to express knowledge is typically finite.

[1] $X \subseteq_f Y$ means X is a finite subset of Y

Definition 7. Let $\mathcal{F} \subset_f \mathcal{L}$ s.t. $\forall \phi \in \mathcal{L}$, there is a unique $\psi \in \mathcal{F}$ s.t. $\phi \equiv \psi$, $\mathtt{Lit}(\phi) = \mathtt{Lit}(\psi)$ and ψ is a CNF formula. The transformation of formulae into the **finite CNF language** is written $\mathtt{CNF}(\phi) = \psi$.

Concrete formulae in the examples are assumed to belong to $\mathcal{F}$.

Let $\sqcup(\Phi)$ represent the **compilation** of Φ as a set of clauses. Intuitively, each formula can be viewed as a set of clauses with an associated sequence of quantifiers. A set of formulae is thus a set of clauses and a sequence of quantifiers, with variables renamed to prevent ambiguities. As an example, assume $\phi_1 = \exists x P(x) \wedge Q(x)$ and $\phi_2 = \exists x Q(x) \vee R(x)$. We have $\sqcup(\{\phi_1, \phi_2\}) = \exists x, x' \{P(x), Q(x), Q(x') \vee R(x')\}$. Formally, for $\Phi = \{\mathcal{Q}_{\phi_j} \phi_j \mid j \in \mathbb{N}\} \subseteq_f \mathcal{F}$, where ϕ_j is a non-quantified CNF formula (i.e., a set of clauses $\mathtt{CNF}(\psi)$ for some $\psi \in \mathcal{F}$), and $\mathcal{Q}_{\phi_j}$ is the sequence of quantifiers associated with ϕ_j, we define $\sqcup(\Phi) = (\mathcal{Q}^*_{\phi_1} \dots \mathcal{Q}^*_{\phi_n}, \bigcup_{\phi \in \Phi} \bigcup_{\delta \in \phi} \delta^*)$, where a renaming is applied to each clause (δ^*) and each sequence of quantifiers ($\mathcal{Q}^*_{\phi_j}$) in order to guarantee that no variable is shared between quantifiers $\mathcal{Q}^*_{\phi_j}$ and $\mathcal{Q}^*_{\phi_k}$ (with $j \neq k$) or between clauses coming from different formulae ϕ_j and ϕ_k (with $j \neq k$). We write $\sqcup(\phi)$ instead of $\sqcup(\{\phi\})$, for $\phi \in \mathcal{F}$.

Throughout the paper, we consider $\sqcup(\Phi)$ as a set with a single formula, where the sequence of quantifiers is the concatenation of $\mathcal{Q}^*_{\phi_1} \dots \mathcal{Q}^*_{\phi_n}$ and the non-quantified part is the CNF formula corresponding to the set of clauses $\bigcup_{\phi \in \Phi} \bigcup_{\delta \in \phi} \delta^*$.

For instance, $\sqcup(\{\forall x \exists y\ P(x,y), \forall x\ Q_1(x) \vee Q_2(x)\}) = \sqcup(\{\forall x_1 \exists x_2\ P(x_1, x_2) \wedge \forall x_3\ Q_1(x_3) \vee Q_2(x_3)\}) = \{\forall x_1 \exists x_2 \forall x_3\ \{P(x_1,x_2), Q_1(x_3) \vee Q_2(x_3)\}\}$.

Let us now introduce the notion of compiled argument.

Definition 8. The **compiled argument** of $A \in \mathtt{Arg}(\mathcal{L})$ is $A^{\mathtt{c}} = \langle \sqcup(\mathtt{S}(A)), \mathtt{C}(A) \rangle$. The power "c" denotes the compiled version of an argument. The set of **compiled instantiated arguments** of A is $\mathtt{CI}(A) = \{A^{\mathtt{ci}} = \langle \sqcup(\mathtt{S}(A^{\mathtt{i}})), \mathtt{C}(A^{\mathtt{i}}) \rangle : A^{\mathtt{i}} \in \mathtt{Inst}_{\mathbf{I_{St}}}(A)\}$, where their compiled instantiated supports (resp. claims) are $I_{\mathtt{S}(A)} = \{\mathtt{S}(A^{\mathtt{ci}}) : A^{\mathtt{ci}} \in \mathtt{CI}(A)\}$ (resp. $I_{\mathtt{C}(A)} = \{\sqcup(\mathtt{C}(A^{\mathtt{ci}})) : A^{\mathtt{ci}} \in \mathtt{CI}(A)\}$).

Example 4. From Example 3, the compilation of $A_3^{\mathtt{i}}$ and $B_1^{\mathtt{i}}$ are: $A_3^{\mathtt{ci}} = \langle \{F(t), D(t), \neg F(t) \vee \neg D(t) \vee L(t), F(u), D(u), \neg F(u) \vee \neg D(u) \vee L(u)\}, F(t) \wedge L(t) \wedge F(u) \wedge L(u) \rangle$;
$B_1^{\mathtt{ci}} = \langle \{F(t), D(t), \neg F(t) \vee \neg D(t) \vee L(t)\}, F(t) \wedge L(t) \rangle$. Moreover, consider $C \in \mathtt{Arg}(\mathcal{L})$ such that $C = \langle \{P(a) \wedge Q(a) \wedge Q(b)\}, P(a) \wedge Q(a) \rangle$, its compilation is: $C^{\mathtt{c}} = \langle \{P(a), Q(a), Q(b)\}, P(a) \wedge Q(a) \rangle$.

The compilation allows us to capture the similarity between $F(t) \wedge D(t)$ and $F(t), D(t)$. Moreover, we can see that the compilation $C^{\mathtt{c}}$ is not concise, as it includes irrelevant information ($Q(b)$) for its claim. As shown in [15], clausal arguments ensure conciseness.

Definition 9. Two arguments $A, B \in \mathtt{Arg}(\mathcal{L})$ are **equivalent**, denoted by $A \approx B$, iff there is a bijection $f : I_{\mathtt{S}(A)} \to I_{\mathtt{S}(B)}$ s.t. $\forall \Phi \in I_{\mathtt{S}(A)}$, $\Phi \cong f(\Phi)$, and $f' : I_{\mathtt{C}(A)} \to I_{\mathtt{C}(B)}$ s.t. $\forall \Psi \in I_{\mathtt{C}(A)}$, $\Psi \cong f'(\Psi)$. Otherwise, $A \not\approx B$ (not equivalent).

Definition 10. Let $A, B \in \mathtt{Arg}(\mathcal{L})$, A is a **sub-argument** of B, denoted $A \sqsubseteq B$, if there is a bijection $f : I_{\mathtt{S}(A)} \to I_{\mathtt{S}(B)}$ s.t. $\forall \Phi \in I_{\mathtt{S}(A)}$, $\Phi \subseteq f(\Phi)$.

3. Principles for Similarity Measures on FOL Arguments

A similarity measure indicates whether two objects (e.g., predicates, or arguments) share some features.

Definition 11. For $\mathbb{X}$ a set of objects, a **similarity measure** on $\mathbb{X}$ is a function $\mathtt{sim}\mathbb{X} : \mathbb{X} \times \mathbb{X} \to [0, 1]$.

In this section, we focus on similarity measures over arguments, *i.e.*, $\mathbb{X} = \mathtt{Arg}(\mathcal{L})$, assuming the existence of $\mathtt{sim}\mathbb{P}$ (resp. $\mathtt{sim}\mathbb{C}$), a similarity measures on predicates (resp. on constants). Intuitively, $\mathtt{simArg}(A, B)$ is close to 0 if the difference between A and B is significant, and close to 1 if they are similar. Several principles that similarity measures should satisfy have been discussed in the literature [9, 16]. Some of the principles (Maximality, Symmetry, Substitution, and Syntax Independence) can be stated exactly as in [15], since they do not concern the internal structure of the arguments. Notice that some authors have argued against the fact that a similarity measures should absolutely satisfy symmetry [17, 18]. The others must be adapted to our FOL-based arguments.

Principle 1 states that arguments sharing no content should have a similarity of 0. While in propositional logic, identifying common atoms is enough, here we must consider predicates and constants. In the context of quantified predicates, we instantiate the variables to analyse the constants. The next three principles consider only compiled instantiated arguments with no irrelevant information (first condition), ensuring safe handling of their similarity. In Principle 1, the second condition excludes the case where both arguments have an empty support and so no intersection to compare. The third (resp. fourth) condition ensures that the arguments have no similarity between the predicates and constants appearing in their support (resp. claim).

Principle 1. Let $\mathbf{I_{St}}$ an interpretation over a structure $\mathbf{St}$, a similarity measure simArg satisfies **Minimality** iff $\forall A, B \in \mathtt{Arg}(\mathcal{L})$, if $\forall A^{\mathtt{ci}} \in \mathtt{CI}(A), \forall B^{\mathtt{ci}} \in \mathtt{CI}(B)$:

1. $A^{\mathtt{ci}}, B^{\mathtt{ci}} \in \mathtt{Arg}(\mathcal{L})$,
2. A and B are not trivial,
3. $\forall \phi \in \mathtt{S}(A^{\mathtt{ci}}),\ \forall l_A \in \mathtt{Lit}(\phi),\ \forall \psi \in \mathtt{S}(B^{\mathtt{ci}}),\ \forall l_B \in \mathtt{Lit}(\psi),\ \mathtt{sim}\mathbb{P}(l_A, l_B) = 0,\ \forall c_A \in \mathtt{Para}(\phi),$ $\forall c_B \in \mathtt{Para}(\psi),$
   $\mathtt{sim}\mathbb{C}(c_A, c_B) = 0,$
4. $\forall \phi \in \sqcup(\mathtt{C}(A^{\mathtt{ci}})),\ \forall l_A \in \mathtt{Lit}(\phi),\ \forall \psi \in \sqcup(\mathtt{C}(B^{\mathtt{ci}})),$
   $\forall l_B \in \mathtt{Lit}(\psi),\ \mathtt{sim}\mathbb{P}(l_A, l_B) = 0,\ \forall c_A \in \mathtt{Para}(\phi),$
   $\forall c_B \in \mathtt{Para}(\psi),\ \mathtt{sim}\mathbb{C}(c_A, c_B) = 0,$

then $\mathtt{simArg}(A, B) = 0$.

The second (resp. third) principle states that the more an argument shares formulae in its support (resp. claim) with an another one, the higher is their similarity.

The compound nature of arguments (support and claim) prevents guarantees when both are considered. Thus, the second condition neutralizes one part to focus on the other. For Principle 2 focusing on supports we ensure that we have identical or completely different claims. As for Principle 3, we ensure that the support are the same, otherwise with completely different support we will have also completely different claim. The third condition guarantees that supports (resp. claims) of A and B share more elements than those of A and C. The fourth condition indicates that the more distinct elements you have, the less similar you are. The fifth condition ensures the reliability of conditions 3 and 4 by guaranteeing that distinct elements of A have no similarity with B and C.

Principle 2. Let $\mathbf{I_{St}}$ an interpretation over a structure $\mathbf{St}$, a similarity measure simArg satisfies **Monotony** iff $\forall A, B, C \in \mathtt{Arg}(\mathcal{L})$, if $\forall A^{\mathtt{ci}} \in \mathtt{CI}(A), \forall B^{\mathtt{ci}} \in \mathtt{CI}(B), \forall C^{\mathtt{ci}} \in \mathtt{CI}(C)$:

1. $A^{\mathtt{ci}}, B^{\mathtt{ci}}, C^{\mathtt{ci}} \in \mathtt{Arg}(\mathcal{L})$,
2. $\sqcup(\mathtt{C}(A^{\mathtt{ci}})) = \sqcup(\mathtt{C}(B^{\mathtt{ci}}))$ or
   $\forall \phi \in \sqcup(\mathtt{C}(A^{\mathtt{ci}})),\ \forall l_A \in \mathtt{Lit}(\phi),\ \forall \psi \in \sqcup(\mathtt{C}(C^{\mathtt{ci}})),$
   $\forall l_C \in \mathtt{Lit}(\psi),\ \mathtt{sim}\mathbb{P}(l_A, l_C) = 0,\ \forall c_A \in \mathtt{Para}(\phi),$
   $\forall c_C \in \mathtt{Para}(\psi),\ \mathtt{sim}\mathbb{C}(c_A, c_C) = 0,$
3. $\mathtt{S}(A^{\mathtt{ci}}) \cap \mathtt{S}(C^{\mathtt{ci}}) \subseteq \mathtt{S}(A^{\mathtt{ci}}) \cap \mathtt{S}(B^{\mathtt{ci}})$,
4. $\mathtt{S}(B^{\mathtt{ci}}) \setminus \mathtt{S}(A^{\mathtt{ci}}) \subseteq \mathtt{S}(C^{\mathtt{ci}}) \setminus \mathtt{S}(A^{\mathtt{ci}})$, and
5. for $\overline{A} = \mathtt{S}(A^{\mathtt{ci}}) \setminus \mathtt{S}(B^{\mathtt{ci}}),\ \forall \phi \in \overline{A},\ \forall l_{\overline{A}} \in \mathtt{Lit}(\phi),\ \forall \psi \in \mathtt{S}(B^{\mathtt{ci}}) \cup \mathtt{S}(C^{\mathtt{ci}}),\ \forall l_{BC} \in \mathtt{Lit}(\psi),$ $\mathtt{sim}\mathbb{P}(l_{\overline{A}}, l_{BC}) = 0, \forall c_{\overline{A}} \in \mathtt{Para}(\phi), \forall c_{BC} \in \mathtt{Para}(\psi), \mathtt{sim}\mathbb{C}(c_{\overline{A}}, c_{BC}) = 0,$

then $\mathtt{simArg}(A, B) \geq \mathtt{simArg}(A, C)$.
• (**Strict Monotony**) If the inclusion in condition 3. is strict or, $\mathtt{S}(A^{\mathtt{ci}}) \cap \mathtt{S}(C^{\mathtt{ci}}) \neq \emptyset$ and the inclusion in condition 4. is strict, then $\mathtt{simArg}(A, B) > \mathtt{simArg}(A, C)$.

Principle 3. Let $\mathbf{I_{St}}$ an interpretation over a structure $\mathbf{St}$, a similarity measure $\mathtt{simArg}$ satisfies **Dominance** iff $\forall A, B, C \in \mathtt{Arg}(\mathcal{L})$, if $\forall A^{\mathtt{ci}} \in \mathtt{CI}(A), \forall B^{\mathtt{ci}} \in \mathtt{CI}(B), \forall C^{\mathtt{ci}} \in \mathtt{CI}(C)$:

1. $A^{\mathtt{ci}}, B^{\mathtt{ci}}, C^{\mathtt{ci}} \in \mathtt{Arg}(\mathcal{L})$,
2. $\mathtt{S}(B^{\mathtt{ci}}) = \mathtt{S}(C^{\mathtt{ci}})$,
3. $\sqcup(\mathtt{C}(A^{\mathtt{ci}})) \cap \sqcup(\mathtt{C}(C^{\mathtt{ci}})) \subseteq \sqcup(\mathtt{C}(A^{\mathtt{ci}})) \cap \sqcup(\mathtt{C}(B^{\mathtt{ci}}))$,
4. $\sqcup(\mathtt{C}(B^{\mathtt{ci}})) \setminus \sqcup(\mathtt{C}(A^{\mathtt{ci}})) \subseteq \sqcup(\mathtt{C}(C^{\mathtt{ci}})) \setminus \sqcup(\mathtt{C}(A^{\mathtt{ci}}))$, and
5. for $\overline{A} = \sqcup(\mathtt{C}(A^{\mathtt{ci}})) \setminus \sqcup(\mathtt{C}(B^{\mathtt{ci}})), \forall \phi \in \overline{A}, \forall l_{\overline{A}} \in \mathtt{Lit}(\phi), \forall \psi \in \sqcup(\mathtt{C}(B^{\mathtt{ci}})) \cup \sqcup(\mathtt{C}(C^{\mathtt{ci}})),\ \forall l_{BC} \in \mathtt{Lit}(\psi)$, $\mathtt{sim}\mathbb{P}(l_{\overline{A}}, l_{BC}) = 0,\ \ \forall c_{\overline{A}} \in \mathtt{Para}(\phi),\ \ \forall c_{BC} \in \mathtt{Para}(\psi)$, $\mathtt{sim}\mathbb{C}(c_{\overline{A}}, c_{BC}) = 0$,

then $\mathtt{simArg}(A, B) \geq \mathtt{simArg}(A, C)$.
• (**Strict Dominance**) If the inclusion in condition 3. is strict or, $\sqcup(\mathtt{C}(A^{\mathtt{i}})) \cap \sqcup(\mathtt{C}(C^{\mathtt{i}})) \neq \emptyset$ and the inclusion in condition 4. is strict, then $\mathtt{simArg}(A, B) > \mathtt{simArg}(A, C)$.

4. Similarity Models

Defining similarity between arguments involves analyzing multiple levels of their CNF structure. At each level, we provide an abstract definition of similarity measure using the previous level, followed by a possible instantiation.
Level 1: Compare pairs of literals (Section 4.1).
Level 2: Aggregate the similarity of pairs of literals to compare grounded clauses (Section 4.2).
Level 3: Combine the similarity of pairs of grounded clauses to compare instantiated supports or claims (Section 4.3).
Level 4: Compute the similarity between sets of instantiations, for each instantiated argument (Section 4.4).
The overall similarity between two arguments is obtained by combining the similarities of their supports and claims obtained from Level 4.

4.1. Similarity between literals

Definition 12. A **similarity measure between two literals** is a function $\mathtt{sim}\mathbb{L} : \mathbb{L} \times \mathbb{L} \to [0, 1]$.

We now instantiate this abstract definition by computing similarity between literals via atom comparison (ignoring polarity), then adjusting scores based on polarity. In this instance, atom similarity depends on two factors: constant vectors and predicate values, with the functions "**c**" applying $\mathtt{sim}\mathbb{C}$ on each pair of constants and "**p**" which is a direct application of $\mathtt{sim}\mathbb{P}$ on the pair of predicates. Both are then combined using an aggregation function "**g**".

Definition 13. Let $\mathbf{c} : \bigcup_{j,k=1}^{+\infty} \mathbb{C}^j \times \mathbb{C}^k \to [0, 1]$ be a similarity measure between a pair of vectors of constants, $\mathbf{p} : \mathbb{P} \times \mathbb{P} \to [0, 1]$ be a similarity measure between a pair of predicates and $\mathbf{g} : [0, 1] \times [0, 1] \to [0, 1]$ be an aggregation function. Given two atoms $\alpha_1 = P_1(a_1, \ldots, a_j)$ and $\alpha_2 = P_2(b_1, \ldots, b_k)$, to compute the **similarity score between atoms** α_1 and α_2 we define $\mathtt{sim}\mathbb{A}^{\langle \mathbf{g},\mathbf{p},\mathbf{c} \rangle} : \mathbb{A} \times \mathbb{A} \to [0, 1]$ s.t. $\mathtt{sim}\mathbb{A}^{\langle \mathbf{g},\mathbf{p},\mathbf{c} \rangle}(\alpha_1, \alpha_2) = \mathbf{g}\Big(\mathbf{p}(\mathtt{Pred}(\alpha_1), \mathtt{Pred}(\alpha_2)), \mathbf{c}\big(\mathtt{Para}(\alpha_1), \mathtt{Para}(\alpha_2)\big)\Big)$.

A possible **p** is the function returning 1 if the predicates are the same, 0 otherwise. Let x, y be two arbitrary objects. The **equality function** $\mathtt{eq} : \mathbb{X} \times \mathbb{X} \to \{0, 1\}$ is defined by $\mathtt{eq}(x, y) = 1$ if $x = y$; or $\mathtt{eq}(x, y) = 0$ otherwise. We propose an instance of **c** suited to vectors of constants.

Definition 14. Let $X = \langle x_1, \ldots, x_j \rangle, Y = \langle y_1, \ldots, y_k \rangle$ be vectors of constants, and $\mathtt{sim}\mathbb{C}$ a similarity measure between constants. The **pointwise similarity** between X and Y is:

$$\mathtt{pws}_{\mathtt{sim}\mathbb{C}}(X, Y) = \begin{cases} 1 & X = Y = \langle\rangle \\ \frac{\sum_{i=1}^{\mathtt{min}(j,k)} \mathtt{sim}\mathbb{C}(x_i, y_i)}{\mathtt{max}(j,k)} & \text{otherwise} \end{cases}$$

In this paper, we only use $\texttt{pws}_{\texttt{eq}}$, which we simplify as $\texttt{pws}$.

We use polarities as binary criteria to determine whether two atoms can be considered similar.

Definition 15. Consider two literals $l_1, l_2 \in \mathbb{L}$, such that the respective atoms are α_1 and α_2. We define $\texttt{sim}\mathbb{L}^{\langle \mathbf{g},\mathbf{p},\mathbf{c}\rangle} : \mathbb{L} \times \mathbb{L} \to [0,1]$, the **similarity measure between two literals** according to a similarity measure between atoms $\texttt{sim}\mathbb{A}^{\langle \mathbf{g},\mathbf{p},\mathbf{c}\rangle}$ such that: $\texttt{sim}\mathbb{L}^{\langle \mathbf{g},\mathbf{p},\mathbf{c}\rangle}(l_1, l_2) =$

$$\begin{cases} \texttt{sim}\mathbb{A}^{\langle \mathbf{g},\mathbf{p},\mathbf{c}\rangle}(\alpha_1, \alpha_2) & \text{if } \texttt{Pol}(l_1) = \texttt{Pol}(l_2) \\ 0 & \text{otherwise} \end{cases}$$

Let $\texttt{avg}$ the average function, we denote $\texttt{sim}\mathbb{L}^{\langle \texttt{avg},\texttt{eq},\texttt{pws}\rangle}$ (resp. $\texttt{sim}\mathbb{A}^{\langle \texttt{avg},\texttt{eq},\texttt{pws}\rangle}$) by $\texttt{sL}$ (resp. $\texttt{sA}$).

Example 5. $\texttt{sL}(P(a), \neg P(a)) = 0$ as they have opposite polarities, while $\texttt{sL}(P(a,b), P(a,c)) = \frac{3}{4}$: $\texttt{sA}(P(a,b), P(a,c)) = \texttt{avg}(\texttt{eq}(P,P), \texttt{pws}(\langle a,b\rangle, \langle a,c\rangle)) = \texttt{avg}(1, \frac{\texttt{eq}(a,a)+\texttt{eq}(b,c)}{2}) = \texttt{avg}(1, \frac{1}{2}) = \frac{3}{4}$.

We could use more sophisticated measures between literals, e.g. assigning a similarity score of 1 to semantically opposite predicates with opposite polarity, e.g., $\neg even(k)$ and $odd(k)$. For this, $\texttt{sim}\mathbb{C}$ and $\texttt{sim}\mathbb{P}$ should consider semantics rather than just syntax. Concretely, the function $\texttt{eq}$ (in this work $\texttt{sim}\mathbb{P} = \texttt{sim}\mathbb{C} = \texttt{eq}$), could be replaced by a semantic similarity, e.g., using NLP techniques, such as cosine similarity between word embeddings (like Word2Vec [19], GloVe [20], or BERT-based models [21]).

4.2. Similarity between grounded clauses

From levels 2 to 4, we use membership functions to assess an object's similarity to a set.

Definition 16. Given $\mathbb{X}$ a set of objects, $x \in \mathbb{X}$ an object, $X \subseteq \mathbb{X}$, $\oplus$ an aggregation function and $\texttt{sim}\mathbb{X}$ a similarity measure on $\mathbb{X}$, the **membership function** of x in X, $\varepsilon^{\oplus}_{\texttt{sim}\mathbb{X}} : \mathbb{X} \times 2^{\mathbb{X}} \to [0,1]$ is defined by: $\varepsilon^{\oplus}_{\texttt{sim}\mathbb{X}}(x, X) = \oplus_{x' \in X}(\texttt{sim}\mathbb{X}(x, x'))$.

The classical set-membership can be captured by $\varepsilon^{\texttt{max}}_{\texttt{eq}}$. Now we can evaluate how much a literal is similar to a clause, *i.e.* a set of literals. We define the function $\texttt{s}\mathbb{L} = \texttt{sim}\mathbb{L}^{\langle \mathbf{g},\mathbf{p},\mathbf{c}\rangle}$.

Definition 17. Let $l \in \mathbb{L}$ be a literal, $L \subseteq \mathbb{L}$ be a set of literals. We define the **membership of a literal** in a set of literals by the function $\varepsilon^{\texttt{avg}}_{\texttt{s}\mathbb{L}} : \mathbb{L} \times 2^{\mathbb{L}} \to [0,1]$ such that:

$$\varepsilon^{\texttt{avg}}_{\texttt{s}\mathbb{L}}(l, L) = \texttt{avg}_{l' \in L}(\texttt{sim}\mathbb{L}^{\langle \mathbf{g},\mathbf{p},\mathbf{c}\rangle}(l, l')).$$

To compare a literal with a set of literals in a clause, we choose to use average aggregation instead of max, as unlike conjunction, disjunction does not guarantee that the max similarity is meaningful. Then, the similarity between two grounded clauses is computed by $\texttt{sim}\mathbb{G}^{\varepsilon^{\texttt{avg}}_{\texttt{s}\mathbb{L}}}$.

Definition 18. A **similarity measure between two grounded clauses** is a function $\texttt{sim}\mathbb{G}^{\varepsilon^{\texttt{avg}}_{\texttt{s}\mathbb{L}}} : 2^{\mathbb{L}} \times 2^{\mathbb{L}} \to [0,1]$.

In Def. 18, we define the similarity between two grounded clauses using a similarity measure (as in Def. 11), based on the membership function $\varepsilon^{\texttt{avg}}_{\texttt{s}\mathbb{L}}$ to compare literals with sets of literals (i.e., grounded clauses). Def. 18 outlines the general concept of grounded clause similarity. In this paper, we use Tversky's ratio model [17] as a concrete approach, though other methods meeting the requirements of Def. 18 and 11 could also be used. Tversky's ratio model [17] is a general family of similarity measures that includes well-known measures like [22], [23], [24], [25], and [26]. We extend it by using our parametrizable membership function ε (see Def. 16) instead of standard set membership operators.

Symmetric Measures	Non-Symmetric Measures
$\mathtt{Tve}^{1,1,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{jac}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$	$\mathtt{Tve}^{1,0,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{ns\text{-}jac}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$
$\mathtt{Tve}^{0.5,0.5,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{dic}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$	$\mathtt{Tve}^{0.5,0,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{ns\text{-}dic}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$
$\mathtt{Tve}^{0.25,0.25,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{sor}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$	$\mathtt{Tve}^{0.25,0,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{ns\text{-}sor}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$
$\mathtt{Tve}^{0.125,0.125,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{adb}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$	$\mathtt{Tve}^{0.125,0,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{ns\text{-}adb}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$
$\mathtt{Tve}^{2,2,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{ss}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$	$\mathtt{Tve}^{2,0,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{ns\text{-}ss}^{\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y)$

Table 1
Parametric (non-)symmetric similarity measures.

Definition 19. Let $X, Y \subseteq \mathbb{X}$ be arbitrary sets of objects, $\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}$ be a membership function with $\oplus$ an aggregation function and $\mathtt{sim}\mathbb{X}$ a similarity measure on $\mathbb{X}$. Let a be the *average of the total similarity* between the elements of X with respect to Y, and the elements of Y with respect to X; and b (resp. c) be the *total dissimilarities* of the elements of X (resp. Y) with respect to Y (resp. X), defined as follows:

- $a = \mathtt{avg}\Big(\sum_{x\in X} \varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}(x,Y), \sum_{y\in Y} \varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}(y,X)\Big)$,
- $b = \sum_{x\in X}(1 - \varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}(x,Y))$,
- $c = \sum_{y\in Y}(1 - \varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}(y,X))$.

Let two coefficients $\alpha, \beta \in [0, +\infty)$, the **extended Tversky measure** between X and Y is:

$$\mathtt{Tve}^{\alpha,\beta,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \begin{cases} 1 & \text{if } X = Y \\ \frac{a}{a+(\alpha\times b)+(\beta\times c)} & \text{otherwise} \end{cases}$$

Classical similarity measures (see Table 1 in [9] for definitions) can be derived with $\alpha = \beta = 2^{-n}$, using classical set-membership. Specifically, the Jaccard measure (jac) corresponds to $n = 0$, Dice (dic) to $n = 1$, Sorensen (sor) to $n = 2$, Anderberg (adb) to $n = 3$, and Sneath and Sokal (ss) to $n = -1$. Moreover, when $\alpha = \beta$ the Tversky measure is symmetric.

Proposition 1. For any sets of objects $X, Y \subseteq \mathbb{X}$, any $\alpha \in [0, +\infty)$, and any membership function $\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}$, we have $\mathtt{Tve}^{\alpha,\alpha,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(X,Y) = \mathtt{Tve}^{\alpha,\alpha,\varepsilon^{\oplus}_{\mathtt{sim}\mathbb{X}}}(Y,X)$.

Table 1 presents the parametric (non-)symmetric extensions of well-known similarity measures, including Jaccard, Dice, Sorensen, Anderberg, and Sokal and Sneath.

These measures satisfy intuitive properties: in the symmetric case, sets are maximally similar if they are identical; in the non-symmetric case, if one is included in the other.

Proposition 2. Let $\alpha \in [0, +\infty)$, $X, Y \subseteq \mathbb{X}$, and $\mathtt{sim}\mathbb{X}$ respects the maximality property: $\forall x \in \mathbb{X}, \mathtt{sim}\mathbb{X}(x,x) = 1$.
• If $Y = X$, then $\mathtt{Tve}^{\alpha,\alpha,\varepsilon^{\mathtt{max}}_{\mathtt{sim}\mathbb{X}}}(X,Y) = 1$ (symmetric case).
• If $X \subseteq Y$, then $\mathtt{Tve}^{\alpha,0,\varepsilon^{\mathtt{max}}_{\mathtt{sim}\mathbb{X}}}(X,Y) = 1$ (non-symmetric).

Example 6. Let $P_1 = P(d,e)$, $P_2 = P(d,f)$, $Q = Q(e,d)$.
Let $\mathtt{sG} = \mathtt{jac}^{\varepsilon^{\mathtt{avg}}_{\mathtt{sL}}}$, then $\mathtt{sG}(\mathtt{Lit}(P_1), \mathtt{Lit}(P_2 \vee Q)) = \frac{a}{a+b+c} \approx 0.23$ where:
$a = \mathtt{avg}(\varepsilon^{\mathtt{avg}}_{\mathtt{sL}}(P_1, \{P_2, Q\}), \varepsilon^{\mathtt{avg}}_{\mathtt{sL}}(P_2, \{P_1\}) + \varepsilon^{\mathtt{avg}}_{\mathtt{sL}}(Q, \{P_1\}))$
$= \mathtt{avg}(\mathtt{avg}(\frac{3}{4}, 0), \frac{3}{4} + 0) = 0.5625;$
$b = 1 - \varepsilon^{\mathtt{avg}}_{\mathtt{sL}}(P_1, \{P_2, Q\}) = 1 - \mathtt{avg}(\frac{3}{4}, 0) = 0.625;$
$c = (1 - \varepsilon^{\mathtt{avg}}_{\mathtt{sL}}(P_2, \{P_1\})) + (1 - \varepsilon^{\mathtt{avg}}_{\mathtt{sL}}(Q, \{P_1\})) = 1.25.$

4.3. Similarity between sets of grounded clauses

Let $\mathbb{G}$ be the set of all grounded clauses in FOL. We define the function $\mathtt{s}\mathbb{G} = \mathtt{sim}\mathbb{G}^{\varepsilon^{\mathtt{avg}}_{\mathtt{s}\mathbb{L}}}$.

Definition 20. Let $\delta \in \mathbb{G}$ and $\Delta \subseteq \mathbb{G}$. We define the **membership of a grounded clause** in a set of grounded clauses by the function $\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{G}} : \mathbb{G} \times 2^{\mathbb{G}} \to [0, 1]$ such that:

$$\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{G}}(\delta, \Delta) = \mathtt{max}_{\delta' \in \Delta}(\mathtt{s}\mathbb{G}(\delta, \delta')).$$

Definition 21. A **similarity measure between two sets of grounded clauses**, i.e., two instantiations, is a function $\mathtt{sim}\mathbb{I}^{\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{G}}} : 2^{\mathbb{G}} \times 2^{\mathbb{G}} \to [0, 1]$.

We revisit Example 6, but we consider a conjunction instead of the disjunction of P_2 and Q.

Example 7. Let $\mathtt{sG} = \mathtt{jac}^{\varepsilon^{\mathtt{avg}}_{\mathtt{sL}}}$ and $\mathtt{sI} = \mathtt{jac}^{\varepsilon^{\mathtt{max}}_{\mathtt{sG}}}$, then $\mathtt{sI}(\{P_1\}, \{P_2, Q\}) = \frac{a}{a+b+c} = \frac{1}{3}$ where:
$a = \mathtt{avg}(\varepsilon^{\mathtt{max}}_{\mathtt{sG}}(P_1, \{P_2, Q\}), \varepsilon^{\mathtt{max}}_{\mathtt{sG}}(P_2, \{P_1\}) + \varepsilon^{\mathtt{max}}_{\mathtt{sG}}(Q, \{P_1\}))$
$= \mathtt{avg}(\mathtt{max}(\frac{3}{4}, 0), \frac{3}{4} + 0) = 0.75;$
$b = 1 - \varepsilon^{\mathtt{max}}_{\mathtt{sG}}(P_1, \{P_2, Q\}) = 1 - \mathtt{max}(\frac{3}{4}, 0) = 0.25;$
$c = (1 - \varepsilon^{\mathtt{max}}_{\mathtt{sG}}(P_2, \{P_1\})) + (1 - \varepsilon^{\mathtt{max}}_{\mathtt{sG}}(Q, \{P_1\})) = 1.25.$

4.4. Similarity between instantiations

Let $\mathbb{I}$ be the set of all instantiations in FOL which are sets of sets of grounded clauses, i.e., $\mathbb{I} = 2^{2^{\mathbb{G}}}$, and $\mathtt{s}\mathbb{I} = \mathtt{sim}\mathbb{I}^{\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{G}}}$.

Definition 22. Let $\Phi \in \mathbb{I}$ and $I \subseteq \mathbb{I}$. We define the **membership of an instantiation** in a set of instantiations by the function $\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{I}} : \mathbb{I} \times 2^{\mathbb{I}} \to [0, 1]$ such that:

$$\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{I}}(\Phi, I) = \mathtt{max}_{\Phi' \in I}(\mathtt{s}\mathbb{I}(\Phi, \Phi')).$$

Definition 23. A **similarity measure between two sets of instantiations** is a function $\mathtt{s}\mathbb{S} = \mathtt{sim}\mathbb{S}^{\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{I}}} : 2^{\mathbb{I}} \times 2^{\mathbb{I}} \to [0, 1]$.

We call a **similarity model** (SM) any tuple $\mathbf{M} = \langle \mathtt{s}\mathbb{L}, \mathtt{s}\mathbb{G}, \mathtt{s}\mathbb{I}, \mathtt{s}\mathbb{S} \rangle$ defining the 4 levels of similarity. Reexamine arguments A and B, focusing on their compiled claim instantiations from Example 3.

Example 8. Let the set of compiled instantiations of claim of A and B, $I_{\mathtt{C}(A)} = \{\Phi_1 = \{F(t), L(t)\}, \Phi_2 = \{F(u), L(u)\}, \Phi_3 = \{F(t), L(t), F(u), L(u)\}\}$ and $I_{\mathtt{C}(B)} = \{\Psi = \{F(t), L(t)\}\}$. Let the Jaccard similarity model $\mathbf{M}\mathtt{jac} = \langle \mathtt{sL}, \mathtt{sG}, \mathtt{sI}, \mathtt{sS} = \mathtt{jac}^{\varepsilon^{\mathtt{max}}_{\mathtt{sI}}} \rangle$.
$\mathtt{sS}(I_{\mathtt{C}(A)}, I_{\mathtt{C}(B)}) = \frac{a}{a+b+c} = \frac{1.625}{1.625+0.75+0} = \frac{1.625}{2.375} \approx 0.684$, where:
$a = \mathtt{avg}\big(\mathtt{sI}(\Phi_1, \Psi) + \mathtt{sI}(\Phi_2, \Psi) + \mathtt{sI}(\Phi_3, \Psi), \mathtt{max}(\mathtt{sI}(\Psi, \Phi_1), \mathtt{sI}(\Psi, \Phi_2), \mathtt{sI}(\Psi, \Phi_3))\big)$
$= \mathtt{avg}(1 + 0.5 + 0.75,\ 1) = \mathtt{avg}(2.25,\ 1) = \frac{2.25+1}{2} = 1.625$
$b = (1 - \mathtt{sI}(\Phi_1, \Psi)) + (1 - \mathtt{sI}(\Phi_2, \Psi)) + (1 - \mathtt{sI}(\Phi_3, \Psi))$
$= (1 - 1) + (1 - 0.5) + (1 - 0.75) = 0 + 0.5 + 0.25 = 0.75$
$c = 1 - \mathtt{max}(\mathtt{sI}(\Psi, \Phi_1), \mathtt{sI}(\Psi, \Phi_2), \mathtt{sI}(\Psi, \Phi_3)) = 1 - 1 = 0$

Using similarity models, we extend the work of [9] to assess the similarity between two FOL arguments.

Definition 24. Let a coefficient $0 < \eta < 1$, a similarity model $\mathbf{M} = \langle \mathtt{s}\mathbb{L}, \mathtt{s}\mathbb{G}, \mathtt{s}\mathbb{I}, \mathtt{s}\mathbb{S} \rangle$ and $\mathbf{I_{St}}$ an interpretation over a structure $\mathbf{St}$. Let $A, B \in \mathtt{Arg}(\mathcal{L})$, we define the **similarity between FOL arguments** from $\mathtt{Arg}(\mathcal{L}) \times \mathtt{Arg}(\mathcal{L})$ to $[0, 1]$ by $\mathtt{simArg}^{\mathbf{M},\eta}_{\mathbf{I_{St}}}(A, B) =$

$$\eta \times \mathtt{s}\mathbb{S}(I_{\mathtt{S}(A)}, I_{\mathtt{S}(B)}) + (1 - \eta) \times \mathtt{s}\mathbb{S}(I_{\mathtt{C}(A)}, I_{\mathtt{C}(B)}).$$

	$\texttt{simArg}^{\mathbf{M}x}$	$\texttt{simArg}^{\mathbf{Mns}-x}$
Maximality	•	•
Symmetry	•	◦
Substitution	•	◦
Syntax Independence	•	•
Minimality	•	•
Monotony	•	•
Strict Monotony	•	◦
Dominance	•	•
Strict Dominance	•	◦

Table 2
Principles satisfaction, with $x \in \{\texttt{jac}, \texttt{dic}, \texttt{sor}, \texttt{adb}, \texttt{ss}\}$ (• for satisfaction, and ◦ for violation).

Example 9. We conclude the example of A and B. Let $\mathbf{M}\texttt{jac} = \langle \texttt{sL}, \texttt{sG}, \texttt{sI}, \texttt{sS} \rangle$. $\texttt{CI}(A) =$
$\{A_1^{\texttt{ci}} = \langle\{F(t), D(t), \neg F(t) \vee \neg D(t) \vee L(t)\}, F(t) \wedge L(t)\rangle,$
$A_2^{\texttt{ci}} = \langle\{F(u), D(u), \neg F(u) \vee \neg D(u) \vee L(u)\}, F(u) \wedge L(u)\rangle,$
$A_3^{\texttt{ci}} = \langle\{F(t), D(t), \neg F(t) \vee \neg D(t) \vee L(t), F(u), D(u), \neg F(u) \vee \neg D(u) \vee L(u)\}, F(t) \wedge L(t) \wedge F(u) \wedge L(u)\rangle\}$;
$\texttt{CI}(B) = \{B_1^{\texttt{ci}} = \langle\{F(t), D(t), \neg F(t) \vee \neg D(t) \vee L(t)\}, F(t) \wedge L(t)\rangle\}$.

As computed in Example 8, $\texttt{sS}(I_{\texttt{S}(A)}, I_{\texttt{S}(B)}) = \frac{1.6875}{2.3125} \approx 0.730$, hence:

$$\texttt{simArg}_{\mathbf{I_{St}}}^{\mathbf{M}\texttt{jac},\frac{1}{2}}(A, B) = \frac{1}{2} \times \frac{1.6875}{2.3125} + \frac{1}{2} \times \frac{1.625}{2.375} = \frac{1}{2} \times 0.730 + \frac{1}{2} \times 0.684 \approx 0.707$$

5. Axiomatic Evaluation

We present the notion of well-behaved similarity model where $\texttt{s}\mathbb{L} = \texttt{sim}\mathbb{L}^{\langle \mathbf{g}, \mathbf{p}, \mathbf{c} \rangle}$, linking the lower-level properties of measures (e.g., Tversky measures) to the higher-level properties of argument similarity.

Definition 25. A similarity model $\mathbf{M} = \langle \texttt{s}\mathbb{L} = \texttt{sim}\mathbb{L}^{\langle \mathbf{g}, \mathbf{p}, \mathbf{c} \rangle}, \texttt{s}\mathbb{G}, \texttt{s}\mathbb{I}, \texttt{s}\mathbb{S} \rangle$ is **well-behaved** iff:

1. a) i. $\mathbf{g}(1, 1) = 1$,
   ii. $\mathbf{g}(0, 0) = 0$,
   b) $\mathbf{p}(P, P) = 1$,
   c) i. $\mathbf{c}(\langle a_1, \dots, a_k \rangle, \langle a_1, \dots, a_k \rangle) = 1$,
   ii. let $a_1, \dots, a_k, b_1, \dots, b_n \in \mathbb{C}$ s.t. $k > 0$, and $n > 0$, if $\forall i \in \{1, \dots, k\}, \forall j \in \{1, \dots, n\}$, $\texttt{sim}\mathbb{C}(a_i, b_j) = 0$, then $\mathbf{c}(\langle a_1, \dots, a_k \rangle, \langle b_1, \dots, b_n \rangle) = 0$,
2. Given $\mathbb{X}$ a set of objects,
   a) $\texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X, X) = 1$ for any set of objects $X \subseteq \mathbb{X}$,
   b) if $\forall x \in X, \forall x' \in X'$, $\texttt{s}(x, x') = 0$ then $\texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X, X') = 0$,
   c) let $X_0, X_1, X_2 \subseteq \mathbb{X}$ s.t. $X_1 \subset X_2$ and $X_2 \setminus X_1 = \{x_2\}$. If $\exists x_0 \in X_0$ s.t. $\texttt{s}(x_0, x_2) = \texttt{s}(x_2, x_0) = 1$ then $\texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X_0, X_2) \geq \texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X_0, X_1)$,
   d) let $X_0, X_1, X_2 \subseteq \mathbb{X}$ s.t. $X_1 \subset X_2$ and $X_2 \setminus X_1 = \{x_2\}$. If $\forall x_0 \in X_0$, $\texttt{s}(x_0, x_2) = \texttt{s}(x_2, x_0) = 0$ then $\texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X_0, X_1) \geq \texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X_0, X_2)$.

Where $\texttt{sim}\mathbb{C}$ is the similarity between constants in $\mathbf{c}$ from $\texttt{s}\mathbb{L}$. In the item 2. $\mathbb{X}$ may denote literals ($\mathbb{L}$), grounded clauses ($\mathbb{G}$), instantiations ($\mathbb{I}$) or set of instantiations ($\mathbb{S}$).

Theorem 1. For any similarity model $\mathbf{M}$, if $\mathbf{M}$ is well-behaved then $\texttt{simArg}_{\mathbf{I_{St}}}^{\mathbf{M},\eta}$ satisfies the following principles: Maximality, Minimality, Monotony and Dominance.

Theorem 2. Let a well-behaved similarity model $\mathbf{M}$.
• $\texttt{simArg}_{\mathbf{I_{St}}}^{\mathbf{M},\eta}$ satisfies Symmetry (resp. Syntax Independence) if $\texttt{s}\mathbb{L}, \texttt{s}\mathbb{G}, \texttt{s}\mathbb{I}, \texttt{s}\mathbb{S}$ are symmetric (resp. syntax independent).
• $\texttt{simArg}_{\mathbf{I_{St}}}^{\mathbf{M},\eta}$ satisfies Strict Monotony and Strict Dominance if it satisfies 2.(c') and 2.(d'), where 2.(c') (resp. 2.(d')) extends 2.(c) (resp. 2.(d)) by adding the condition $\texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X_0, X_1) < 1$ (resp. $\texttt{sim}\mathbb{X}^{\varepsilon_{\texttt{s}}^{\oplus}}(X_0, X_1) > 0$) to deduce their conclusion strictly (i.e., $>$ instead of $\geq$).

Interestingly, some results from the propositional framework in [9] change in FOL: 1. The characterization linking the satisfaction of axioms to a similarity of 1 iff arguments are equivalent is lost. 2. Argument compilation makes equivalent arguments identical, giving them a similarity of 1 solely via Maximality. 3. Logically non-equivalent arguments (e.g., with $\neg even(k)$ and $odd(k)$) can also have a similarity of 1 if deemed similar by a similarity measure. 4. Substitution is no longer implied by other axioms, as the characterization of similarity of 1 for equivalent arguments no longer holds.

The functions $\mathbf{g}$, $\mathbf{p}$ and $\mathbf{c}$ of this paper are well-behaved.

Lemma 1. For $\mathbf{g} \in \{\mathtt{min}, \mathtt{avg}\}$, $\mathbf{p} = \mathtt{eq}$ and $\mathbf{c} = \mathtt{pws}_{\mathtt{eq}}$, $\langle \mathbf{g}, \mathbf{p}, \mathbf{c} \rangle$ satisfies item 1. of Def. 25.

Similar results hold for the Tversky measures used to define $\mathtt{sim}\mathbb{G}^{\varepsilon^{\mathtt{avg}}_{\mathtt{sL}}}$, $\mathtt{sim}\mathbb{I}^{\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{G}}}$, and $\mathtt{sim}\mathbb{S}^{\varepsilon^{\mathtt{max}}_{\mathtt{s}\mathbb{I}}}$, as detailed in Table 1.

Lemma 2. If in $\mathtt{Tve}^{\alpha,\beta,\varepsilon^{\mathtt{max}}_{\mathtt{sim}\mathbb{X}}}$, $\mathtt{sim}\mathbb{X}$ is either $\mathtt{sim}\mathbb{L}^{\langle \mathbf{g},\mathbf{p},\mathbf{c} \rangle}$ s.t. $\langle \mathbf{g}, \mathbf{p}, \mathbf{c} \rangle$ satisfies item 1. of Def. 25, or a similarity measure satisfying item 2. of Def. 25, then $\mathtt{Tve}^{\alpha,\beta,\varepsilon^{\mathtt{max}}_{\mathtt{sim}\mathbb{X}}}$ satisfies item 2. of Def. 25.

Proposition 3. Let $\mathtt{simArg}^{\mathbf{M}_x,\eta}_{\mathbf{I_{St}}}$ with $\eta \in (0,1)$, $x \in \{\mathtt{jac}, \mathtt{dic}, \mathtt{sor}, \mathtt{adb}, \mathtt{ss}, \mathtt{ns\text{-}jac}, \mathtt{ns\text{-}dic}, \mathtt{ns\text{-}sor}, \mathtt{ns\text{-}adb}, \mathtt{ns\text{-}ss}\}$ and $\mathbf{M}_x = \langle \mathtt{sL}, \mathtt{sG}x = x^{\varepsilon^{\mathtt{avg}}_{\mathtt{sL}}}, \mathtt{sI}x = x^{\varepsilon^{\mathtt{max}}_{\mathtt{sG}x}}, \mathtt{sS}x = x^{\varepsilon^{\mathtt{max}}_{\mathtt{sI}x}} \rangle$. All principles are compatible according to Table 2.

The symmetric extended Tversky measures returns a similarity of 1 iff the arguments are equivalent. While non-symmetric measures also show full similarity in another specific case of sub-arguments.

Theorem 3. Let $A, B \in \mathtt{Arg}(\mathcal{L})$ and a similarity model $\mathbf{M}x$ (resp. $\mathbf{M}\mathtt{ns} - x$) from Table 2. For any $\eta \in (0,1)$, we have: $\mathtt{simArg}^{\mathbf{M},\eta}_{\mathbf{I_{St}}}(A, B) = 1$ iff $A \approx B$; and $\mathtt{simArg}^{\mathbf{M}\mathtt{ns}-x,\eta}_{\mathbf{I_{St}}}(A, B) = 1$ iff $A \sqsubseteq B$ and $\sqcup(\mathtt{C}(A)) \subseteq \sqcup(\mathtt{C}(B))$.

Asymmetric measures violate Strict principles and Substitution by scoring 1 for sub-arguments, preventing similarity increases for equivalent arguments and allowing differing argument structures to share a similarity of 1.

6. Related Work

Several works have investigated similarity in logic-based or structured representation systems, but they differ significantly from our work w.r.t. expressivity, the nature of similarity, and intended applications.

Ramon and Bruynooghe [27] define a distance between first-order logic objects, focusing on the structural comparison of clauses and terms. Their approach operates within the syntax of FOL, without addressing quantified formulas explicitly, and relies on clause-level comparisons in normalized form. Similarly, Horváth et al. [28] define recursive similarity functions over first-order terms for instance-based learning, focusing on term structure rather than complete formulae. Their approach does not handle quantifiers or logical connectives. In contrast, our framework operates on FOL formulae with quantifiers and connectives, and supports similarity computation over structured arguments.

In contrast to syntactic or clause-based distances, Williamson [29] introduces a formal logic of comparative similarity using a ternary predicate $S(x, y, z)$, interpreted as "x is more similar to y than to z". While formally elegant, this qualitative approach lacks numerical scores and aggregation.

Similarity in ontology-based or object-oriented systems was studied in [30, 31]. Bisson [30] studies similarity based on hierarchies and attributes in object-oriented representations, but without a logical foundation. Ehrig et al. [31] propose a flexible multi-dimensional framework for ontology matching, but without schema-level comparisons, nor extension to FOL formulae or structured argumentation.

In the description logic (DL) community, several works attempt to define similarity between concepts. Borgida et al. [32] and Janowicz [33] propose semantically grounded measures for DLs, typically based on model-theoretic subsumption or concept overlap. However, DLs are restricted fragments of FOL: they do not support general quantification, nested terms, or logical connectives. In contrast, our framework is designed to operate on first-order logic, enabling a more expressive and flexible similarity computation.

González-Calero et al. [34] explore the use of description logics in case-based reasoning by representing cases as DL concepts and retrieving them via subsumption. While semantically grounded, their approach is limited to concept-level similarity and does not support the composition of similarity measures across multiple structural levels.

In summary, prior work relies either on syntactic distances over terms or clauses, qualitative reasoning frameworks without quantitative evaluation, or semantic similarity restricted to limited fragments such as description logics. In contrast, our contribution defines similarity directly over full first-order logic, supporting quantification and nested connectives, and provides a parametric multi-level model that evaluates similarity both at different structural levels within FOL formulae and within argument structures.

7. Conclusion

This paper introduces parametric similarity models, defining families of similarity measures for FOL arguments, including generalized versions of existing measures (i.e., Tversky) and, for the first time in logical argumentation, non-symmetric measures. Three extended principles are proposed, with well-behaved properties ensuring the satisfaction of some principles. Symmetric Tversky-based measures satisfy all principles, while non-symmetric ones satisfy a subset.

This work opens new research avenues, as exploring additional measures (e.g., semantic similarity with BERT-based models [21]) and principles (e.g., independent distribution [35]) to improve similarity accuracy. Future work will focus on real-world applications, implementing similarity models, and analyzing their complexity.

Acknowledgments

This work was supported by the *Agence Nationale de la Recherche* (ANR) under the projects AGGREEY ANR-22-CE23-0005, AIDAL ANR-22-CPJ1-0061-01, and the Plan d'Investissement France 2030, as part of the Initiative d'Excellence d'Université Côte d'Azur under the reference ANR-15-IDEX-01.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] Q. Zhong, X. Fan, X. Luo, F. Toni, An explainable multi-attribute decision model based on argumentation, Expert Sys. and Appl. 117 (2019) 42–61.
[2] K. Cyras, A. Rago, E. Albini, P. Baroni, F. Toni, Argumentative XAI: A survey, in: Proc. of IJCAI 2021, 2021, pp. 4392–4399.
[3] Y. Guo, T. Yu, L. Bai, J. Tang, Y. Ruan, Y. Zhou, Argumentative explanation for deep learning: A survey, in: International Conference on Unmanned Systems (ICUS), IEEE, 2023, pp. 1738–1743.
[4] B. Irwin, A. Rago, F. Toni, Forecasting argumentation frameworks, arXiv:2205.11590 (2022).
[5] D. Gorur, A. Rago, F. Toni, Argucast: A system for online multi-forecasting with gradual argumentation, in: Arg. & App.@KR'23, CEUR-WS.org, 2023, pp. 40–51.
[6] J. Ben-Naim, V. David, A. Hunter, An axiomatic study of the evaluation of enthymeme decoding in weighted structured argumentation, arXiv preprint arXiv:2411.04555 (2024).
[7] J. Ben-Naim, V. David, A. Hunter, An axiomatic study of a modular evaluation of enthymeme decoding in weighted structured argumentation, in: KR'25, 2025.

[8] V. David, A. Hunter, A logic-based framework for decoding enthymemes in argument maps involving implicitness in premises and claims, in: IJCAI'25, 2025.
[9] L. Amgoud, V. David, Measuring similarity between logical arguments, in: KR'18, 2018, pp. 98–107.
[10] L. Amgoud, V. David, An adjustment function for dealing with similarities, in: COMMA'20, 2020, pp. 79–90.
[11] L. Amgoud, V. David, A General Setting for Gradual Semantics Dealing with Similarity, in: AAAI'21, 2021.
[12] V. David, J. Delobelle, J.-G. Mailly, Similarity measures between order-sorted logical arguments, in: Journées d'Intelligence Artificielle Fondamentale, 2023.
[13] J. Lang, P. Liberatore, P. Marquis, Propositional independence-formula-variable independence and forgetting, J. Artif. Intell. Res. 18 (2003) 391–443.
[14] P. Besnard, A. Hunter, A logic-based theory of deductive arguments, Artificial Intelligence 128 (2001) 203–235.
[15] L. Amgoud, V. David, Similarity measures based on compiled arguments, in: ECSQARU'21, 2021, pp. 32–44.
[16] L. Amgoud, V. David, D. Doder, Similarity measures between arguments revisited, in: ECSQARU'19, 2019, pp. 3–13.
[17] A. Tversky, Features of similarity, Psychological Review 84 (1977) 327–352.
[18] K. P. Jantke, Nonstandard concepts of similarity in case-based reasoning, in: Information Systems in Data Analysis: Prospects – Foundations – Applications, 1994, pp. 28–43.
[19] K. W. Church, Word2vec, Natural Language Engineering 23 (2017) 155–162.
[20] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: EMNLP'14, 2014, pp. 1532–1543.
[21] N. Reimers, Sentence-bert: Sentence embeddings using siamese bert-networks, arXiv preprint arXiv:1908.10084 (2019).
[22] P. Jaccard, Nouvelles recherches sur la distributions florale, Bulletin de la societe Vaudoise des sciences naturelles 37 (1901) 223–270.
[23] L. Dice, Measures of the amount of ecologic association between species, Ecology 26 (1945) 297–302.
[24] T. Sørensen, A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons, Biol. Skr. 5 (1948) 1–34.
[25] M. Anderberg, Cluster analysis for applications. monographs and textbooks on probability and mathematical statistics, 1973.
[26] P. Sneath, R. Sokal, Numerical taxonomy. The principles and practice of numerical classification., 1973.
[27] J. Ramon, M. Bruynooghe, A framework for defining distances between first-order logic objects, in: ILP'98, 1998, pp. 271–280.
[28] T. Horváth, S. Wrobel, U. Bohnebeck, Relational instance-based learning with lists and terms, Machine Learning 43 (2001) 53–80.
[29] T. Williamson, First-order logics for comparative similarity, Notre Dame J. Formal Log. 29 (1988).
[30] G. Bisson, Why and how to define a similarity measure for object based representation systems, Towards Very Large Knowledge Bases (1995) 236–246.
[31] M. Ehrig, P. Haase, M. Hefke, N. Stojanovic, Similarity for ontologies-a comprehensive framework (2005).
[32] A. Borgida, T. Walsh, H. Hirsh, et al., Towards measuring similarity in description logics., Description Logics 147 (2005) 1–8.
[33] K. Janowicz, Sim-dl: Towards a semantic similarity measurement theory for the description logic alcnr in geographic information retrieval, in: OTM 2019 Conferences, 2006, pp. 1681–1692.
[34] P. González-Calero, B. Díaz-Agudo, M. Gómez-Albarrán, et al., Applying dls for retrieval in case-based reasoning, in: DL'99, 1999.
[35] V. David, Dealing with Similarity in Argumentation, Ph.D. thesis, Univ. Toulouse III, 2021.