Token Space: A Category Theory Framework for AI Computations*

Wuming Pan
panwuming@scu.edu.cn

College of Computer Science, Sichuan University, Chengdu, P.R. China, 610065

Abstract

This paper introduces the Token Space framework, a novel mathematical construct designed to enhance the interpretability and effectiveness of deep learning models through the application of category theory. By establishing a categorical structure at the Token level, we provide a new lens through which AI computations can be understood, emphasizing the relationships between tokens, such as grouping, order, and parameter types. We explore the foundational methodologies of the Token Space, detailing its construction, the role of construction operators and initial categories, and its application in analyzing deep learning models, specifically focusing on attention mechanisms and Transformer architectures. The integration of category theory into AI research offers a unified framework to describe and analyze computational structures, enabling new research paths and development possibilities. Our investigation reveals that the Token Space framework not only facilitates a deeper theoretical understanding of deep learning models but also opens avenues for the design of more efficient, interpretable, and innovative models, illustrating the significant role of category theory in advancing computational models.

Keywords: Token Space, Category Theory, Deep Learning, Attention Mechanisms, Transformer Models, Computational Models, AI Computations

1 Introduction

During the last sixty years, category theory has become a tangible and profoundly impactful force in reshaping the journey of mathematical discourse. It has served as a vital foundational tool that enhances our understanding of the complex web of relationships between diverse mathematical ideas and structures. Category theory proves to be one of the essential building blocks for modern mathematics [1] [2]. The consideration of computational structures through the prism of category theory, especially within the frameworks of machine learning and deep learning , brings a novel, structurally abundant angle. This approach expands current theoretical frameworks, enabling novel research paths and development possibilities. One of the examples is Optics, which is possibly one of the more characteristic example consumers of category theory. It describes modular access and modification of separate parts of such data structures , which is necessary for managing large-scale solutions or structures that should not be altered after creation. Deep learning contains other category theory-based aspects alongside Optics, such as functors, natural transformations, and limits that improve existing methods of dealing with mathematical objects and provide new angles of research and understanding.

Category theory shifts the focus from the objects themselves to the relationships between objects as defined by morphisms (structure-preserving maps), offering a unified descriptive framework for diverse structures. This shift from objects to relationships is significantly meaningful for

*This is a draft, completed on April 11, 2024.

computational models in deep learning. Optics, as one of the tools in category theory, is particularly suitable for deep learning because they can express the composability and data transformation processes between layers of neural networks. This allows complex neural network architectures to be managed and optimized through more abstract structures.

Furthermore, tools from category theory, including Optics, elegantly handle processes like forward and backward propagation in deep learning. These processes fundamentally involve complex mappings of data and gradient flows in high-dimensional spaces. Through the framework of category theory, we can construct and understand these processes in a unified and compositional manner, which is crucial for developing new deep learning algorithms and theories.

However, understanding and interpreting deep learning models remains a challenge, partly because we lack a unified and formalized method to describe and analyze their structure. This paper introduces the concept of Token Space, aiming to fill this gap by providing a new mathematical framework for deep learning. Token Space is a framework built on the foundation of category theory, defining Tokens and their combinations and sequences as basic building blocks to construct a complex category with universal expressiveness. This approach allows us to view layers within neural networks as mappings between combinations of Tokens, offering a new way to understand the structure and behavior of deep learning models. Through this formalized representation, we can not only gain insights into the complex relationships between network layers but also explore new areas of network design, optimize existing architectures, and enhance the interpretability and generalizability of models.

The introduction of Token Space represents an innovative integration between category theory and the field of machine learning. It provides a common language and framework for understanding across two seemingly unrelated disciplines, opening new directions for research and potentially guiding us toward more efficient, interpretable machine learning models. This paper will detail the theoretical foundations of Token Space, explore its application in constructing and analyzing deep learning models, and anticipate its potential value and challenges in future research.

By combining the theory and practice of deep learning models, Token Space not only offers a new perspective for current machine learning research but also paves the way for interdisciplinary collaboration and theoretical innovation, demonstrating the significant role of category theory in advancing computational models. We believe the concept of Token Space will become a key tool in understanding and designing the next generation of machine learning models.

Intuitively, the Token Space framework seems particularly well-suited for representing attention mechanisms and Transformer architectures. The core of attention mechanisms and Transformer models is their ability to dynamically assign different importance weights to different parts of the input data and integrate and process information based on these weights. This process of dynamic weighting and information integration aligns closely with the way Token Space framework handles combinations and sequences of Tokens.

In Token Space, Tokens can be seen as atomic units of data or information, while the combinations and sequences of Tokens reflect the structure and relationships among data. By combining and recombining Tokens, it is possible to flexibly express and model the complex interactions between data, which is precisely what attention mechanisms excel at. For instance, Transformer models calculate the relationship weights between different Tokens through self-attention mechanisms, and then integrate and weight the information based on these weights to capture the inherent structure and dependencies of the input data.

The Token Space framework provides a formalized representation method for these dynamic relationships and weight distributions, allowing us to understand and analyze the workings of attention mechanisms from a more abstract level. By defining appropriate operations and mappings within Token Space, it is possible to precisely describe and simulate the flow and processing of information in Transformer models, thus offering strong theoretical support for in-depth study and optimization of such models.

In summary, the integration of category theory, exemplified by tools such as Optics, into machine and deep learning research fosters a deeper theoretical comprehension and spurs the innovation of algorithms. Optics underline the significance of category theory in deep learning's theoretical exploration, offering a versatile toolkit for computational structures. Simultaneously, the Token Space framework's alignment with attention mechanisms and Transformer models illuminates its capability to articulate and scrutinize advanced models, paving the way for novel insights and more potent variations. This approach not only enhances theoretical understanding but also facilitates practical applications, leading to deep learning models that are more interpretable, efficient, and innovative.

This paper is structured to systematically explore the Token Space and its implications for AI computations. Section 2, "How is the Token Space Constructed?", lays the foundational methodologies and theoretical underpinnings of the Token Space. Following this, Section 3, "Construction Operators and Initial Categories", delves into the specifics of construction operators and initial categories, which include Identity Set Categories, Products of Categories, Isomorphism between Categories, and Subsets Extension of Subcategories of Set. Section 4 broadens our discussion to Elementary Token Space and Token Topoi, emphasizing the Token Space.

In Section 5, we turn our attention to "Representing Categories of Structured Objects in Token Space", probing into how structured object categories are represented within the Token Space. Section 6 explores the realm of Token Categories, setting the stage for a deeper investigation into Interior Structure Mapping and Tree Token Classes in Section 7, where we detail the Generation of Tree Tokens and Tokens Maps between Tree Token Classes. Section 8, "Exploring Structure Relations of Token Classes", examines the intricate structure relations among Token Classes. The process of Reification of Tree Token Classes is the focus of Section 9, culminating in Section 10, "Conclusion", where we summarize our findings and outline future research directions.

2 How is the Token Space Constructed?

The initiative behind the construction of the Token Space is to imbue AI computations with meanings at the Token level, such as token grouping, order, and parameter types. Without attributing meanings at the Token level, due to the abstract nature of category theory structures, it becomes quite difficult to grasp the implications and characteristics of computations. Additionally, this causes morphisms themselves to lack an internal categorical structure. These factors prevent category theory from being utilized as a tool to thoroughly explore computational properties, keeping the investigation at the surface level of computational structures. However, category theory is a very powerful and flexible tool, and employing category theory concepts more deeply in AI computation research has significant advantages.

To establish a categorical structure at the Token level, this paper employs two fundamental rules:

1. A set consisting only of identity morphisms constitutes a bi-Cartesian closed category.
2. A category formed by the product of multiple bi-Cartesian closed categories is a bi-Cartesian closed category.

Token Space uses the former to assign meanings to tokens while being capable of representing any function as a morphism.

3 Construction Operators and Initial Categories

In exploring the foundational concepts of functions and operations within sets, we introduce some notations that facilitate a deeper understanding of these structures. By viewing functions as sets of correspondences between elements of two sets, we illuminate the dynamic nature of functions as mechanisms that establish directional correspondences between individual elements. This perspective enriches our comprehension of functions as comprehensive structures of correspondences, laying the groundwork for exploring more complex interactions within category theory.

In delving into the structure and interpretation of functions, along with operations on sets, we introduce refined notations that illuminate the concept of functions as rich structures of *correspondences*. Specifically, a function $f : A \to B$ can be envisioned in two complementary ways. On one hand, as a traditional mapping from elements of set A to set B, and on the other, as a set of explicit correspondences between individual elements, expressed as follows:

$$f = \{(x, f(x)) \mid x \in A\}$$
$$f = \{x \mapsto f(x) \mid x \in A\}$$

The first notation $(x, f(x))$ symbolizes the ordered pairs forming the function f, denoting a correspondence between each element x in set A and a unique element $f(x)$ in set B. The second notation $x \mapsto f(x)$ accentuates the dynamic nature of the function as a set of directional correspondences, where x in A is mapped to $f(x)$ in B. This notation underscores the essence of functions as mechanisms that establish a rule for assigning exactly one corresponding element in B to each element x in A, resonating with categorical concepts by highlighting the individual correspondences.

Additionally, when we discuss the combination of two sets A_1 and A_2, we use the term *disjoint union*, denoted by $A_1 \coprod A_2$. The disjoint union of A_1 and A_2 refers to a set that contains all the elements of A_1 and A_2 without any overlap; if an element is in both A_1 and A_2, it will appear in the union set marked to distinguish which set it originated from. This concept is vital in understanding how we can combine different sets while preserving the identity and properties of each element within those sets.

3.1 Identity Set Categories

Let category $\mathbf{C}_*$ be defined within the framework of $\mathbf{Set}_*$ as consisting of all singleton sets as objects. Specifically, each object is a set containing exactly one element. Define $\mathbf{C}_0$ as a category that contains a single object, the singleton set $\{*\}$, making it a full subcategory of $\mathbf{C}_*$.

Proposition 1. *In $\mathbf{C}_*$, any two objects are isomorphic to each other. $\mathbf{C}_0$ serves as a skeleton of $\mathbf{C}_*$, and is equivalent to $\mathbf{C}_*$, suggesting the existence of a fully faithful and essentially surjective functor between $\mathbf{C}_0$ and $\mathbf{C}_*$.*

Proof. Since every object in $\mathbf{C}_*$ is a singleton set, there exists precisely one morphism between any two objects. This morphism establishes an isomorphism by sending the unique element from one object to another, which illustrates that all objects in $\mathbf{C}_*$ are inherently isomorphic to each other.

A *skeleton* of a category is a subcategory that contains one representative of each isomorphism class of objects in the original category. Since $\mathbf{C}_0$ consists of a single object and fully represents the isomorphism class of singleton sets within $\mathbf{C}_*$, it serves as a skeleton of $\mathbf{C}_*$. This implies that every object in $\mathbf{C}_*$ is isomorphic to the single object in $\mathbf{C}_0$, capturing the essential structure of $\mathbf{C}_*$ without redundancy.

The *equivalence* of categories goes beyond the mere existence of isomorphisms between objects. It suggests that there is a fully faithful (preserves morphisms exactly) and essentially surjective (every object in the target category is isomorphic to the image of an object in the source category) functor between the two categories. Thus, $\mathbf{C}_0$ is equivalent to $\mathbf{C}_*$ because there exists such a functor that connects every object in $\mathbf{C}_*$ to the single object in $\mathbf{C}_0$, and vice versa, in a way that preserves the categorical structures. This functorial relationship upholds the equivalence, indicating that, structurally, the categories are "the same" from a categorical perspective.

Therefore, the intrinsic simplicity and symmetry within $\mathbf{C}_*$ lead us to conclude that the relationship between $\mathbf{C}_0$ and $\mathbf{C}_*$ is straightforward, given the concepts of isomorphisms, skeletons, and category equivalences. □

If a category $\mathbf{C}$ contains initial and terminal objects, along with all products, coproducts, and exponentials (also known as function objects or map objects) for any two objects, then $\mathbf{C}$ is a bi-Cartesian closed category, denoted as a *BICCC*. A topos is a special kind of category that possesses all finite limits, a subobject classifier, and all exponentials.

The category $\mathbf{C}_*$, consisting solely of objects isomorphic to the singleton set $\{*\}$, exhibits the structure of a bi-Cartesian closed category. This is because it possesses initial and terminal objects, finite products and coproducts, as well as exponentials, all of which are trivially realized due to the singular nature of its objects, each being isomorphic to $\{*\}$.

Furthermore, $\mathbf{C}_*$ can be considered a topos. It satisfies the criteria for being a topos by having all finite limits, a subobject classifier (which, in this case, is also isomorphic to $\{*\}$), and by being Cartesian closed, a consequence of its bi-Cartesian closed structure and the homogeneity of its objects.

For any functor $f : \mathbf{C}_0 \to \mathbf{Set}$, the image $f(\mathbf{C}_0)$, which comprises a single set with a single arrow—the identity arrow, can form a category that is both bicartesian closed and a topos. In such a category, the single set and its identity arrow fulfill the roles of all required categorical constructs due to the abstract nature of category theory.

For any $E \in \mathbf{Set}$, we define the category $\mathbf{E} = \mathbf{idset}(E)$, called an *identity set* category, which contains only one object E and a single arrow 1_E, the identity morphism on E. Within this category, E serves simultaneously as the initial object, terminal object, product, coproduct, and exponentiation of itself, showcasing the flexibility of categorical constructs in abstract settings.

The diagrams below visually represent these concepts, showing E as fulfilling multiple categorical roles:

$$\begin{tikzcd}
 & E \arrow[dl, "1_E"'] \arrow[d, dashed, "1_E"] \arrow[dr, "1_E"] & \\
E & E \times E \arrow[l, "1_E"] \arrow[r, "1_E"'] & E
\end{tikzcd}$$

In the category $\mathbf{E}$, consisting of a single object E and its identity morphism 1_E, the coproduct of E with itself, denoted $E \coprod E$, simplifies to E. This reflects the abstract nature of category theory, where the specifics of coproducts, products, and other categorical constructs depend on the category's internal structure. Here, the diagram represents E as its own coproduct, with the identity morphism 1_E serving both as the inclusion morphisms into the coproduct and as the universal morphism from the coproduct.

$$
\begin{tikzcd}
E \arrow[r, "1_E"] & E \coprod E \arrow[d, equal] & E \arrow[l, "1_E"'] \\
& E &
\end{tikzcd}
$$

$$
\begin{tikzcd}
& E^E \arrow[d, dashed, "1_E"] \\
E \times E \arrow[ur, "1_E"] \arrow[r, "1_E"'] & E
\end{tikzcd}
$$

$$
\begin{tikzcd}
\Omega_E \arrow[r, "true"] & E \\
E \arrow[u, dashed, "char_E"] \arrow[ur, "1_E"'] &
\end{tikzcd}
$$

In this simplified setting, the only subobject of E is E itself, reinforcing the abstract notion that E can act as its own subobject classifier, a key feature of a topos.

Proposition 2. *Each id-set category, characterized by a single object and its identity morphism, is both bicartesian closed and a topos.*

Proof. Given the abstract definitions of bicartesian closed categories and topoi, an id-set category $\mathbf{E}$ satisfies all necessary criteria through its single object E and identity morphism 1_E. E acts as its own product, coproduct, exponential object, and subobject classifier, meeting the definitions of a bicartesian closed category and a topos in a context where categorical constructs are defined abstractly. $\square$

3.2 Products of Categories

In below text, we regard $\mathcal{U}$ as a operator which can map any categories product $\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n$, where $\mathbf{C}_i$ is a sub category of $\mathbf{Set}$, to a functor

$\mathcal{U}_{\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n} : \mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n \to \mathbf{Set}$

such that $\mathcal{U}_{\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n}$ sends $\langle A_1, A_2, \cdots, A_n \rangle$ to $A_1 \coprod A_2 \coprod \cdots \coprod A_n$ where A_1 is a set obtained by forgetting structure on A_1 and sends arrow $\langle f_1, f_2, \cdots, f_n \rangle$ to $f_1 \coprod f_2 \coprod \cdots \coprod f_n$. When clear from context, we use $\mathcal{U}$ to denote $\mathcal{U}_{\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n}$.

The following lemma can be easily verified.

Lemma 1. *Let $\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_n$ be categories.*

1. If each $\mathbf{C}_i$ is bi-Cartesian closed, then their product category $\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n$ is also bi-Cartesian closed.

2. If each $\mathbf{C}_i$ is a topos, then their product category $\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n$ is also a topos.

In the study of category theory, understanding the structural components of categories and their interactions is pivotal. The table below provides a comparative overview of the structural components for individual categories $\mathbf{C}_i$ and the product category $\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n$. This comparison illuminates the manner in which properties and structures are preserved or transformed under the formation of product categories.

Table 1: Structure Components of $\mathbf{C}_i$ and $\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n$

Category	$\mathbf{C}_i$	$\mathbf{C}_1 \times \mathbf{C}_2 \times \cdots \times \mathbf{C}_n$
Initial object	s_i	$\langle s_1, s_2, \cdots, s_n \rangle$
Terminal object	t_i	$\langle t_1, t_2, \cdots, t_n \rangle$
Product	$A_i \times B_i$	$\langle A_1 \times B_1, A_2 \times B_2, \cdots, A_n \times B_n \rangle$
Coproduct	$A_i \coprod B_i$	$\langle A_1 \coprod B_1, A_2 \coprod B_2, \cdots, A_n \coprod B_n \rangle$
Exponents	$A_i^{B_i}$	$\left\langle A_1^{B_1}, A_2^{B_2}, ..., A_n^{B_n} \right\rangle$
Limit	$\mathbf{Lim}F_i,$	$\langle \mathbf{Lim}F_1, \mathbf{Lim}F_2, ..., \mathbf{Lim}F_n \rangle$
Limiting cone	v_j	$\langle v_1, v_2, ..., v_n \rangle$
Truth object	$\{0,1\}$	$\langle \{0,1\}, \{0,1\}, ..., \{0,1\} \rangle$

This exposition delves into two fundamental operations within the realm of category theory: products and coproducts. Given two sequences of objects from respective categories, $\langle A_1, A_2, \ldots, A_n \rangle$ and $\langle B_1, B_2, \ldots, B_n \rangle$, the operations are defined as follows:

For the *product* operation, we have

$$\langle A_1, A_2, \ldots, A_n \rangle \times \langle B_1, B_2, \ldots, B_n \rangle = \langle A_1 \times B_1, A_2 \times B_2, \ldots, A_n \times B_n \rangle .$$

This equation demonstrates the element-wise application of the product operation on the corresponding objects of two sequences, yielding a new sequence where each element is the product of the respective elements from the original sequences.

Similarly, for the *coproduct* operation, we observe

$$\langle A_1, A_2, \ldots, A_n \rangle \coprod \langle B_1, B_2, \ldots, B_n \rangle = \left\langle A_1 \coprod B_1, A_2 \coprod B_2, \ldots, A_n \coprod B_n \right\rangle .$$

This reveals that, analogous to the product operation, the coproduct operation applied element-wise to two sequences of objects results in a new sequence. Each element of this sequence is the coproduct of the corresponding elements in the original sequences.

Both equations underscore the abstract yet consistent manner in which category theory treats operations on objects, irrespective of whether the operation is a product or a coproduct. This uniform treatment facilitates a deeper understanding of the structures and relationships within and across categories.

The category **Set** is bi-Cartesian closed and is a topos and its construct components are as in table 2. **Set** and identity set category $\mathbf{E} = \mathbf{idset}(E)$ with $E \in ISets$ can be used to construct a product category $\mathbf{Set} \times \mathbf{E}$ with projections

$$\mathbf{Set} \overset{P}{\leftarrow} \mathbf{Set} \times \mathbf{E} \overset{Q}{\rightarrow} \mathbf{E}.$$

Corollary 1. $\mathbf{Set} \times \mathbf{E}$ *is bi-Cartesian closed and a topos.*

In $\mathbf{Set} \times \mathbf{E}$, the initial object, terminal object, product, coproduct, map object of two objects $\langle A, E \rangle, \langle B, E \rangle \in \mathbf{Set} \times \mathbf{E}$, the limit $\underline{\mathbf{Lim}}F$ of functor $F : J \to \mathbf{Set} \times \mathbf{E}$ for a finite category J, and its limiting cone $v : \underline{\mathbf{Lim}}F \to F$ and truth object are as in table 2.

Table 2: Structure Components of $\mathbf{E}$, $\mathbf{Set}$, and $\mathbf{Set} \times \mathbf{E}$

Category	$\mathbf{E}$	$\mathbf{Set}$	$\mathbf{Set} \times \mathbf{E}$
Initial object	E	$\varnothing$	$\langle \varnothing, E \rangle$
Terminal object	E	$\{0\}$	$\langle \{0\}, E \rangle$
Product	$E = E \times E$	$A \times B$	$\langle A \times B, E \rangle$
Coproduct	$E = E \coprod E$	$A \coprod B$	$\langle A \coprod B, E \rangle$
Exponents	$E = E^E$	A^B	$\left\langle A^B, E\} \right\rangle$
Limit	$E = \underline{\mathbf{Lim}} F,$	$\underline{\mathbf{Lim}} F$	$\langle \underline{\mathbf{Lim}} P \circ F, E \rangle$
Limiting cone	$v_j = 1_E$	v_j	$\langle P(v_j), 1_E \rangle$
Truth object	E	$\{0,1\}$	$\langle \{0,1\}, E \rangle$

Proposition 3. *The functor* $\mathcal{U}$:$\mathbf{Set} \times \mathbf{E} \to \mathbf{Set}$ *is injective on objects and arrows.*

Proof. This can be easily verified by the definition of $\mathcal{U}$. ☐

Corollary 2. $\mathbf{Set} \times \mathbf{E}$ *is isomorphic to* $\mathcal{U}(\mathbf{Set} \times \mathbf{E})$. $\mathcal{U}(\mathbf{Set} \times \mathbf{E})$ *is bi-Cartesian closed and is a topos.*

3.3 Isomorphism between Categories

Given categories $\mathbf{S}_1$ and $\mathbf{S}_2$, and a isomorphism $\mathcal{I}$:$\mathbf{S}_1 \to \mathbf{S}_2$. Then one is bi-Cartesian closed or a topos iff another is. The isomorphism preserve the structure components of the two catogories.

Example 1. *Let* $\mathbf{FMon_{Set}}$ *be a category whose objects are free monoids on objects in* $\mathbf{Set}$*, and arrows in* $\mathbf{FMon_{Set}}$ *are monoid homomorphisms extended from functions (arrows) between their generator sets in* $\mathbf{Set}$*. Let* $\mathcal{H}$:$\mathbf{Set} \to \mathbf{FMon_{Set}}$ *be a functor which sends each* A *in* $\mathbf{Set}$ *to free monoid* $\mathcal{H}(A) = \langle A^*, \cdot, \varepsilon \rangle$ *generated on* A *and sends each function* $f : A \to B$ *to monoid homomorphism* $\mathcal{H}(f) = f^*$ *extended from* f*. The underlying set of* $\mathcal{H}(A)$ *are* A^**.* $\mathcal{H}$ *is an isomorphism between* $\mathbf{Set}$ *and* $\mathbf{FMon_{Set}}$*.* $\mathbf{FMon_{Set}}$ *is bi-Cartesian closed and is a topos. In* $\mathbf{FMon_{Set}}$*, the construct components of bi-Cartesian closed category and topos are as in table 3 where* $F : \mathbf{J} \to \mathbf{FMon_{Set}}$ *is a functor for a finite category* $\mathbf{J}$*, and its limiting cone is* $v : \underline{\mathbf{Lim}} F \to F$*. Let* $\mathcal{F} : \mathbf{FMon_{Set}} \to \mathbf{Set}$ *be the forgetful functor which sends each monoid* $\mathcal{H}(A)$ *to its underling set* A^**. Then* $\mathcal{F}$ *is an injective on objects and arrows, and* $\mathbf{FMon_{Set}}$ *is isomorphic to* $\mathcal{F}(\mathbf{FMon_{Set}})$*. It follows that* $\mathcal{F}(\mathbf{FMon_{Set}})$ *is bi-Cartesian closed and is a topos. Notice that* $\mathcal{F}$ *map all structure components in* $\mathcal{H}(\mathbf{Set})$ *to those in* $\mathcal{F}(\mathbf{FMon_{Set}})$*.*

Given a subcategory $\mathbf{Sc}$ of $\mathbf{Set}$ and a isomorphism $\mathcal{I}$:$\mathbf{Set} \to \mathbf{Sc}$. Then $\mathbf{Sc}$ is bi-Cartesian closed and is a topos. Taking the given isomorphism $\mathcal{I}$, then $\mathcal{I}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$ is a subcategory of $\mathbf{Sc}$, hence a subcategory of $\mathbf{Set}$.

Lemma 2. *Given a subcategory* $\mathbf{Sc}$ *of* $\mathbf{Set}$ *and a isomophism* $\mathcal{I}$:$\mathbf{Set} \to \mathbf{Sc}$*. For any subcategory* $\mathbf{S}'$ *of* $\mathbf{Set}$*, there is a isomophism from* $\mathbf{S}'$ *to* $\mathcal{I}\left(\mathbf{S}'\right)$*.*

Proof. Clear. ☐

Corollary 3. *Given a subcategory* $\mathbf{Sc}$ *of* $\mathbf{Set}$ *and a isomophism* $\mathcal{I}$:$\mathbf{Set} \to \mathbf{Sc}$*.* $\mathcal{I}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$ *is bi-Cartesian closed and a topos.*

Table 3 provides a comparison of the structure components between the category $\mathbf{Sc}$ and the functorial image $\mathcal{I}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$, highlighting the transformations applied by the functors $\mathcal{I}$ and $\mathcal{U}$ to the product category $\mathbf{Set} \times \mathbf{E}$.

Table 3: Structure Components of **Sc** and $\mathcal{I}\left(\mathcal{U}(\mathbf{Set}\times\mathbf{E})\right)$

Category	**Sc**	$\mathcal{I}\left(\mathcal{U}\left(\mathbf{Set}\times\mathbf{E}\right)\right)$
Initial object	$\mathcal{I}(\varnothing)$	$\mathcal{I}(\varnothing\coprod E)$
Terminal object	$\mathcal{I}\left(\{0\}\right)$	$\mathcal{I}\left(\{0\}\coprod E\right)$
Product	$\mathcal{I}(A\times B)$	$\mathcal{I}\left((A\times B)\coprod E\right)$
Coproduct	$\mathcal{I}(A\coprod B)$	$\mathcal{I}\left((A\coprod B)\coprod E\right)$
Exponents	$\mathcal{I}\left(A^B\right)$	$\mathcal{I}\left(A^B\coprod E\right)$
Limit	$\mathcal{I}\left(\underline{\mathbf{Lim}}\mathcal{I}^{-1}\circ F\right)$	$\mathcal{I}\left(\underline{\mathbf{Lim}}\left(P\circ\mathcal{U}^{-1}\circ\mathcal{I}^{-1}\circ F\right)\coprod E\right)$
Limiting cone	$\mathcal{I}\left(\mathcal{I}^{-1}(v)\right)$	$\mathcal{I}\left(P\circ\mathcal{U}^{-1}\circ\mathcal{I}^{-1}\left(v_i\right)\coprod 1_E\right)$
Truth object	$\mathcal{I}\left(\{0,1\}\right)$	$\mathcal{I}\left(\{0,1\}\coprod E\right)$

While in $\mathcal{I}\left(\mathcal{U}\left(\mathbf{Set}\times\mathbf{E}\right)\right)$, the construct of bi-Cartesian closed category and topos are as in table 3 where $F:\mathbf{J}\!\to\mathcal{I}\left(\mathcal{U}\left(\mathbf{Set}\times\mathbf{E}\right)\right)$ is a functor for a finite category J, and its limiting cone is $v:\underline{\mathbf{Lim}}F\to F$.

3.4 Subsets Extension of Subcategories of Set

Let **Sc** be a subcategory of **Set**, then $\mathbf{SX_{Sc}}$, the subsets extension of **Sc**, is a category such that:

- Its objects are all sets in **Sc** and all subsets of them. If a set A is a subset of two objects in **Sc**, then there are two objects which copy A are added to $\mathbf{SX_{Sc}}$, and the two objects copying A represent subsets of different objects in **Sc**.
- There is a function $Or:\mathbf{Sc}\to\mathbf{SX_{Sc}}$ such that If $S\in\mathbf{Sc}$, then $Or(S)=S$; If S is added to $\mathbf{SX_{Sc}}$ because S is a subset of a set $A\in\mathbf{Sc}$, then $Or(S)=A$. Each object $S\in\mathbf{SX_{Sc}}$ is given a label $Or(S)\in\mathbf{Sc}$, and S is also denoted as $S_{Or(S)}$. Note that a set S in **Set** may correspond to many objects in $\mathbf{SX_{Sc}}$ which is labeled by different sets of which S is a subset. Especially for each set $A\in\mathbf{Sc}$ there is a set $\varnothing_A\in\mathbf{SX_{Sc}}$.
- Its arrows are all arrows in **Sc**, the inclusion maps $\mathbf{in}(S_1,S_2)$ where $S_1\subset S_2$ and $Or\left(S_1\right)=Or\left(S_2\right)$, and arrows $\mathbf{on}(f,S):S_{Or(S)}\to f\left(S_{Or(S)}\right)$, representing function f restricted on domain $S_{Or(S)}$ and codomain $f\left(S_{Or(S)}\right)$, for each $f:Or(S)\to A$ in **Sc**.
- For any two objects $S_1,S_2\in\mathbf{SX_{Sc}}$, S_1 is regarded as a subobject of S_2, if only if $Or\left(S_1\right)$ is a subobject of $Or\left(S_2\right)$ and with $i:Or\left(S_1\right)\to Or\left(S_2\right)$ the inclusion arrow in **Sc** we have that $i\left(S_1\right)$ is a subset of S_2. In other words, the diagram

$$\begin{tikzcd} Or(S_1) \arrow[r, "i"] & Or(S_2) & S_2 \arrow[l, hook', "\mathbf{in}(S_2,Or(S_2))"'] \\ S_1 \arrow[u, "\mathbf{in}(S_1,Or(S_1))"] \arrow[r, "\mathbf{on}(i,S_1)"] & i(S_1) \arrow[u, "\mathbf{in}(i(S_1),Or(S_2))"'] \arrow[r, "\mathbf{in}(i(S_1),S_2)"] & S_2 \arrow[u, equal] \end{tikzcd}$$

  commute.

Evidently, there is an inclusion functor $\mathbf{I_{Sc}}:\mathbf{Sc}\to\mathbf{SX_{Sc}}$.

Proposition 4. *For any arrow $t:S_1\to S_2$ in $\mathbf{SX_{Sc}}$, there is a unique arrow $t_{Or}:Or\left(S_1\right)\to Or\left(S_2\right)$ in **Sc** such that $t=\mathbf{in}\left(t(S_1),S_2\right)\circ\mathbf{on}(t_{Or},S_1)$.*

Proof. This is simple. ☐

We use $Or(t)$ to denote t_{Or} then Or can be regarded as a functor from $\mathbf{Sc}$ to $\mathbf{SX_{Sc}}$. Note that for any arrow $f : Or\,(S_1) \to Or\,(S_2)$ in $\mathbf{Sc}$, there is a arrow $\mathbf{on}(f, S_1)$ in $\mathbf{SX_{Sc}}$, but there may be not an arrow $t : S_1 \to S_2$ such that $Or(t) = f$. For the convenience, we always adopt the notation

$$\mathbf{on}(f, S_1, S_2) = \mathbf{in}\,(f(S_1), S_2) \circ \mathbf{on}(f, S_1)$$

when $\mathbf{in}\,(f(S_1), S_2)$ exists.

Theorem 1. *If* $\mathbf{Sc}$ *is bi-Cartesian closed, then* $\mathbf{SX_{Sc}}$ *is bi-Cartesian closed. If* $\mathbf{Sc}$ *has all finite limits, then* $\mathbf{SX_{Sc}}$ *has all finite limits.*

Proof. See following paragraphs. ☐

Let $S_1, S_2 \in \mathbf{SX_{Sc}}$, and J be a finite category and $F : J \to \mathbf{SX_{Sc}}$ is a functor. Suppose the structure components in $\mathbf{Sc}$ for $Or\,(S_1)\,, Or\,(S_2) \in \mathbf{Sc}$ and functor $Or \circ F$ are as in table 4, then structure components of $\mathbf{SX_{Sc}}$ are also summarized as in table 4 and explained as follows.

Initial object. There may not an arrow from initial object o_i in $\mathbf{S}$ to an object, which is not $\mathbf{Sc}$ originally, in $\mathbf{SX_{Sc}}$. And for $A, B \in \mathbf{Sc}$, if there is not an arrow from A to B, then there is not an arrow from $\varnothing_A$ to $\varnothing_B$. However, there is an arrow from $\varnothing_{o_i}$ to any an object of $\mathbf{SX_S}$. Hence $\varnothing_{o_i}$ is the initial object of $\mathbf{SX_{Sc}}$.

Terminal object. Terminal object in $\mathbf{SX_{Sc}}$ is the terminal object o_t in $\mathbf{Sc}$.

Limits and Products. Let diagram

$$Or\,(S_1) \xleftarrow{p} Or\,(S_1) \times Or\,(S_2) \xrightarrow{q} Or\,(S_2)$$

be a production diagram in $\mathbf{Sc}$, and for a finite category J and $F : J \to \mathbf{SX_{Sc}}$, the map

$$v : \underline{\mathbf{Lim}} Or \circ F \to Or \circ F$$

is a limiting cone in $\mathbf{Sc}$. Because

$$F(j) \subset Or\,(F(j)) = Or \circ F(j)$$

for any $Or \circ F(j)$, we have

$$\bigcap_{j \in J} v_j^{-1}\,(F(j)) \subset \bigcap_{j \in J} v_j^{-1}\,(Or \circ F(j)) \subset \underline{\mathbf{Lim}} Or \circ F$$

Let $\tau : S \to F$ be a cone over F, then $Or(\tau) : Or(S) \to Or \circ F$ is a cone over $Or \circ F$, and there is a unique arrow $t : Or(S) \to \underline{\mathbf{Lim}} Or \circ F$ make the diagram

$$\begin{tikzcd}
Or(S) \arrow[r, "t"] \arrow[d, "Or(\tau_i)"] \arrow[dr, "Or(\tau_k)"] \arrow[drrr, "Or(\tau_l)"] & \lim Or \circ F \arrow[dl, "v_i"] \arrow[d, "v_k"] \arrow[drr, "v_l"] & & \\
Or \circ F(i) \arrow[r, "Or(F(u))"] & Or \circ F(k) \arrow[r] & ... \arrow[r] & Or \circ F(l)
\end{tikzcd}$$

commute. Because

$$Or\left(\tau_j\right)(S) \subset (F(j))$$

we have

$$S \subset Or\left(\tau_j\right)^{-1}(F(j))\,.$$

And because

$$v_j\left(t\left(Or\left(\tau_j\right)^{-1}(F(j))\right)\right) = Or\left(\tau_j\right)\left(Or\left(\tau_j\right)^{-1}(F(j))\right)$$

we have

$$S \subset_{j\in J} Or\left(\tau_j\right)^{-1}(F(j))$$

and

$$t\left(Or\left(\tau_j\right)^{-1}(F(j))\right) \subset v_j^{-1}\left(v_j\left(t\left(Or\left(\tau_j\right)^{-1}(F(j))\right)\right)\right) = v_j^{-1}(F(j))$$

Therefore, we have

Proposition 5. *Notation being as just used, we have*

$$t(S) \subset t({}_{j\in J}Or\left(\tau_j\right)^{-1}(F(j))) \subset_{j\in J} v_j^{-1}(F(j))\,.$$

Proof. This is simple. □

Let

$$in_F = \mathbf{in}(t(S),_{j\in J} v_j^{-1}(F(j))),$$

and

$$in_j = \mathbf{in}(v_j({}_{j\in J}v_j^{-1}(F(j))), F(j))$$

be the inclusion maps in $\mathbf{SX_{Sc}}$, then diagram

$$\begin{tikzcd}
S \arrow[r, "in_F\circ\mathbf{on}(t,S)"] \arrow[d, "\tau_i"] \arrow[dr, "\tau_k"] \arrow[drrr, "\tau_l"] & {}_{j\in J}v_j^{-1}(F(j)) \arrow[dl, "in_i\circ\mathbf{on}(v_i,-)"] \arrow[d, "in_k\circ\mathbf{on}(v_k,-)"] \arrow[drr, "in_k\circ\mathbf{on}(v_l,-)"] & & \\
F(i) \arrow[r, "F(u)"] & F(k) \arrow[r] & ... \arrow[r] & F(l)
\end{tikzcd}$$

commute. Hence

$$\underline{\mathbf{Lim}}F \cong_{j\in J} v_j^{-1}\left(F(j)\right))$$

Coproducts. Let

$$Or\left(S_1\right) \overset{inc_1}{\to} Or\left(S_1\right) \coprod Or\left(S_2\right) \overset{inc_2}{\leftarrow} Or\left(S_2\right)$$

be a coproduct diagram in **Sc**. Let

$$ins_i = \mathbf{in}(inc_i^{-1}\left(S_i\right), inc_1^{-1}\left(S_1\right) \cup inc_2^{-1}\left(S_2\right))$$

for $i = 1, 2$. Then

$$S_1 \overset{ins_1 \circ \mathbf{on}(inc_1, S_1)}{\to} inc_1^{-1}\left(S_1\right) \cup inc_2^{-1}\left(S_2\right) \overset{ins_2 \circ \mathbf{on}(inc_2, S_2)}{\leftarrow} S_2$$

is a coproduct diagram in $\mathbf{SX_{Sc}}$.

Exponents. For any $S_1, S_2 \in \mathbf{SX_{Sc}}$, there is an arrow *evl* in diagram

$$Or\left(S_1\right)^{Or(S_2)} \times Or\left(S_2\right) \overset{evl}{\to} Or\left(S_1\right)$$

is an evaluation map in **Sc.** And let p_m and q_m in diagram

$$Or\left(S_1\right)^{Or(S_2)} \overset{p_m}{\leftarrow} Or\left(S_1\right)^{Or(S_2)} \times Or\left(S_2\right) \overset{q_m}{\to} Or\left(S_2\right)$$

be production projections in **Sc.**

Lemma 3. *The set*

$$\mathbf{preSet}\left(S_1, S_2\right) = \left\{ S \left| p_m^{-1}(S) \cap q_m^{-1}\left(S_2\right) \subset evl^{-1}\left(S_1\right) \right. \right\}$$

has a supremum element.

Let $\mathbf{preExp}\left(S_1, S_2\right)$ be a subset of $Or\left(S_1\right)^{Or(S_2)}$ such that $\mathbf{preExp}\left(S_1, S_2\right)$ is the supremum element in $\mathbf{preSet}\left(S_1, S_2\right)$.

Proposition 6. *Notation being as just used, we have* $\mathbf{preExp}\left(S_1, S_2\right)$ *is the exponent object* $\left(S_1\right)^{S_2}$.

Proof. For any S_1, S_2, We easily know $\mathbf{preExp}\left(S_1, S_2\right)$ exist. For any $S_3 \in \mathbf{SX_{Sc}}$, if there is an arrow $g : S_3 \times S_2 \to S_1$ then there is an arrow $f : Or\left(S_3\right) \to Or\left(S_1\right)^{Or(S_2)}$ such that the diagram

$$\begin{tikzcd}
Or(S_3) \arrow[d, "f^*"] & Or(S_3) \times Or(S_2) \arrow[l, "p_3"'] \arrow[r, "Or(g)"] \arrow[d, "f^* \times 1_{Or(S_2)}"] & Or(S_1) \arrow[d, equal] \\
Or(S_1)^{Or(S_2)} & Or(S_1)^{Or(S_2)} \times Or(S_2) \arrow[l, "p_m"] \arrow[r, "evl"] & Or(S_1)
\end{tikzcd}$$

commute in **Sc,** where arrows p_3 and p_m are projection maps and $Or(g), f^*$ uniquely determines each other. Because

$$S_3 \times S_2 \subset Or(g)^{-1}(S_1)$$

we have

$$f^* \times 1_{Or(S_2)}(S_3 \times S_2) \subset evl^{-1}(S_1).$$

We have

$$f^* \circ p_3 (S_3 \times S_2) \subset Or(S_1)^{Or(S_2)}$$

and

$$f^* \times 1_{Or(S_2)}(S_3 \times S_2) \subset (f^* \circ p_3 (S_3 \times S_2)) \times S_2.$$

Therefore

$$p_m^{-1}(f^* \circ p_3 (S_3 \times S_2)) \cap q_m^{-1}(S_2) \subset evl^{-1}(S_1)$$

Hence

$$f^* \circ p_3 (S_3 \times S_2) = f^*(S_3) \subset \mathbf{preExp}(S_1, S_2).$$

Similarly, if there is $f : S_3 \to \mathbf{preExp}(S_1, S_2)$, then there is an arrow

$$g^* : Or(S_3) \times Or(S_2) \to Or(S_1)$$

such that the diagram

$$\begin{tikzcd}
Or(S_3) \arrow[d, "Or(f)"] & Or(S_3) \times Or(S_2) \arrow[l, "p_3"'] \arrow[r, "g^*"] \arrow[d, "Or(f)\times 1_{Or(S_2)}"] & Or(S_1) \arrow[d, equal] \\
Or(S_1)^{Or(S_2)} & Or(S_1)^{Or(S_2)} \times Or(S_2) \arrow[l, "p_m"'] \arrow[r, "evl"] & Or(S_1)
\end{tikzcd}$$

commute in **Sc,** where $g^*, Or(f)$ uniquely determines each other. Because

$$Or(f) \circ p_3 (S_3 \times S_2) = Or(f)(S_3) \subset \mathbf{preExp}(S_1, S_2)$$

hence

$$Or(f) \times 1_{Or(S_2)}(S_3 \times S_2)$$

$$\subset \mathbf{preExp}(S_1, S_2) \times S_2$$

$$\subset p_m^{-1}\left(\mathbf{preExp}\left(S_1, S_2\right)\right) \cap q_m^{-1}\left(S_2\right)$$

$$\subset evl^{-1}\left(S_1\right)$$

Therefore

$$g^*\left(S_3 \times S_2\right) \subset S_1$$

and $\mathbf{on}\left(g^*, S_3 \times S_2, S_1\right)$ is the unique arrow determined by f. By now the proposition is proved. $\square$

Proposition 7. *For any $S_1, S_2 \in \mathbf{SX_{Sc}}$, we have*

$$\mathbf{preExp}\left(S_1, S_2\right) = Or\left(S_1\right)^{Or\left(S_2\right)} - p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right).$$

Proof. Because

$$p_m^{-1}\left(Or\left(S_1\right)^{Or\left(S_2\right)} - p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right)\right) \cap q_m^{-1}\left(S_2\right) \subset evl^{-1}\left(S_1\right),$$

then

$$Or\left(S_1\right)^{Or\left(S_2\right)} - p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right) \subset \mathbf{preExp}\left(S_1, S_2\right).$$

Because

$$Or\left(S_1\right)^{Or\left(S_2\right)} - \mathbf{preExp}\left(S_1, S_2\right) \subset p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right)$$

then

$$p_m^{-1}\left(Or\left(S_1\right)^{Or\left(S_2\right)} - \mathbf{preExp}\left(S_1, S_2\right)\right) \subset p_m^{-1}\left(p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right)\right)$$

then

$$Or\left(S_1\right)^{Or\left(S_2\right)} \times Or\left(S_2\right) - p_m^{-1}\left(p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right)\right)$$

$$\subset Or\left(S_1\right)^{Or\left(S_2\right)} \times Or\left(S_2\right) - p_m^{-1}\left(Or\left(S_1\right)^{Or\left(S_2\right)} - \mathbf{preExp}\left(S_1, S_2\right)\right)$$

Therefore

$$p_m\left(Or\left(S_1\right)^{Or\left(S_2\right)} \times Or\left(S_2\right) - p_m^{-1}\left(p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right)\right)\right)$$

$$\subset p_m\left(Or\left(S_1\right)^{Or\left(S_2\right)} \times Or\left(S_2\right) - p_m^{-1}\left(Or\left(S_1\right)^{Or\left(S_2\right)} - \mathbf{preExp}\left(S_1, S_2\right)\right)\right)$$

then

$$Or\left(S_1\right)^{Or(S_2)} - p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right)$$

$$\subset \mathbf{preExp}\left(S_1, S_2\right)$$

Hence the proposition is proved.

□

Table 4 contrasts the structure components of the categories **Sc** and $\mathbf{SX_{Sc}}$, highlighting their differences and similarities in terms of initial and terminal objects, products, coproducts, exponents, limits, and limiting cones.

Table 4: Structure Components of **Sc** and $\mathbf{SX_{Sc}}$

Category	**Sc**	$\mathbf{SX_{Sc}}$
Initial object	o_i	$\varnothing_{o_i}$
Terminal object	o_t	o_t
Product	$Or\left(S_1\right) \times Or\left(S_2\right)$	$S_1 \times S_2 \cong p^{-1}\left(S_1\right) \cap q^{-1}\left(S_2\right)$
Coproduct	$Or\left(S_1\right) \coprod Or\left(S_2\right)$	$S_1 \coprod S_2 \cong$ $inc_1^{-1}\left(S_1\right) \cup inc_2^{-1}\left(S_2\right)$
Exponents	$Or\left(S_1\right)^{Or(S_2)}$	$Or\left(S_1\right)^{Or(S_2)} -$ $p_m\left(q_m^{-1}\left(S_2\right) - evl^{-1}\left(S_1\right)\right)$
Limit	$\underline{\mathbf{Lim}} Or \circ F$	$\underline{\mathbf{Lim}} F \cong_{j \in J} v_j^{-1}\left(F(j)\right)$
Limiting cone	$v : \underline{\mathbf{Lim}} Or \circ F \to Or \circ F$	$\underline{\mathbf{Lim}} F$ to $F(j)$: $in_j \circ v_j \circ$ $\mathbf{in}\left(\underline{\mathbf{Lim}} F, \underline{\mathbf{Lim}} Or \circ F\right)$
Truth object	Ω	Ω?

If we do not know more properties about **Sc**, there seems no category structures in $\mathbf{SX_{Sc}}$ which can classify subobjects in it.

4 Elementary Token Space and Token Topoi

Let

$$\mathcal{H} : \mathbf{Set} \to \mathbf{FMon_{Set}}$$

be a functor which sends each A in **Set** to free monoid $\mathcal{H}(A) = A^*$ generated on A and sends each function $f : A \to B$ to monoid homomorphism $\mathcal{H}(f) = f^*$ extended from f. Let $\mathcal{F}$: $\mathbf{FMon_{Set}} \to \mathbf{Set}$ be the forgetful functor which sends each monoid $\mathcal{H}(A)$ to its underling set $|A^*|$.

Definition 1 (Tokenoid)**.** *Unary operator map elements in $\mathcal{F}\left(A^*\right)$ to tuples. For example*

$$\left(x_1 \cdot x_2 \cdot \cdots \cdot x_n\right) = \left(x_1, x_2, \cdots, x_n\right)$$

The set of Tokens $\left(\mathcal{F}\left(A^*\right)\right)$, which is also denoted as $A^{\&}$, is called a Tokenoid *generated on A. Category*

$$\mathbf{PatA_{Set}} = \left(\mathcal{F}\left(\mathbf{FMon_{Set}}\right)\right)$$

is called pre-Token category*. We will view as a functor form* $\mathcal{F}(\mathbf{FMon_{Set}})$ *to* $\mathbf{PatA_{Set}}$*, which map* $\mathcal{F}(A^*)$ *to* $(\mathcal{F}(A^*))$ *and* $\mathcal{F}(f^*)$ *to* $(\mathcal{F}(f^*))$*.*

Let $\mathcal{M} = \circ\mathcal{F} \circ \mathcal{H}$ be a functor which sends each A in **Set** to Tokenoid $\mathcal{M}(A)$ generated on A and sends each function $f : A \to B$ to Tokenoid homomorphism $\mathcal{M}(f)$ extended from f.

Corollary 4. $\mathcal{M}$ *is an isomorphism between* **Set** *and* $\mathbf{PatA_{Set}}$*. And* $\mathbf{PatA_{Set}}$ *is bi-Cartesian closed and is a topos. And*

$$\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$$

is bi-Cartesian closed and a topos.

While in $\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$, the construct of bi-Cartesian closed category and topos are as in table 2 where

$$F : \mathbf{J} \to \mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$$

is a functor for a finite category J, and its limiting cone is $v : \underline{\mathbf{Lim}}F \to F$.

Category

$$TS_{\mathbf{E}} = \mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{Set}\times\mathbf{E})))}$$

with $\mathbf{E}$ an identity set category is also called an *elementary Token space*, each object $\Upsilon_{Or(\Upsilon)}$ in $TS_{\mathbf{E}}$ with

$$Or(\Upsilon) = \mathcal{M}(\mathcal{G}(\mathcal{U}(\langle B, E\rangle)))$$

where $B \in \mathbf{Set}$ is also represented as

$$\langle B, E, \Upsilon\rangle .$$

Notice that $\Upsilon \subset \mathcal{M}(\mathcal{U}(\langle B, E\rangle)) \in \mathbf{Set}$, each element r in Υ can be regard as a tuple $(x_1, x_2, \cdots, x_n)$ in $(B \coprod E)^{\&}$. A object $T = \langle B, E, \Upsilon\rangle$ in $TS_{\mathbf{E}}$ is called a *Token class*, or *p-class*, in which B, E and Υ, which are denoted as $\mathbf{base}(T)$, $\mathbf{structure}(T)$ and $\mathbf{heap}(T)$ respectively, are called the *base*, *core* and *Token heap*; tuples in Υ are called *Tokens*. Each arrow

$$a : \langle B_1, E, \Upsilon_1\rangle \to \langle B_2, E, \Upsilon_2\rangle$$

in $TS_{\mathbf{E}}$ is called a *Token map*, or *p-map*, and also represented as $\langle f, 1_E, p\rangle$ where $\langle f, 1_E\rangle$ is an arrow in $\mathbf{Set} \times \mathbf{E}$ and

$$p = \mathbf{in}(\mathcal{M}(\mathcal{U}(\langle f, 1_E\rangle))(\langle B_1, E, \Upsilon_1\rangle), \mathcal{U}(\langle B_2, E\rangle))\circ$$

$$\mathbf{on}(\mathcal{M}(\mathcal{U}(\langle f, 1_E\rangle)), \mathcal{U}(\langle B_1, E\rangle)))$$

Notice that

$$\mathcal{M}(\mathcal{U}(\langle f, 1_E\rangle))$$

is homomorphism between free Tokenoid

$$\mathcal{M}(\mathcal{U}(\langle B_1, E\rangle))$$

and

$$\mathcal{M}(\mathcal{U}(\langle B_2, E\rangle))$$

, and $f \coprod 1_E$ is an arrow in $\mathcal{U}(\mathbf{Set} \times \mathbf{E})$. Therefore, for any $(x_1, x_2, \cdots, x_n) \in \Upsilon$, there is

$$p\left(\left(x_1, x_2, \cdots, x_n\right)\right)=\left(f \coprod 1_E\left(x_1\right), f \coprod 1_E\left(x_2\right), \cdots, f \coprod 1_E\left(x_n\right)\right)$$

Because p is uniquely determined by f in $\langle f, 1_E, p\rangle$, when clear from context we use f to $\langle f, 1_E, p\rangle$ and p. The set $\hom\left(T_1, T_2\right)$ denotes the set of all Tokens maps between T_1 and T_2.

The arrows in

$$TS_{\mathbf{E}}=\mathbf{SX}_{\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))}$$

are either the Token algebra homomorphisms or the partial maps of them. Notice that the inclusion maps are the partial maps of identity Token algebra homomorphisms.

Table 5 delineates the structure components of $\mathbf{SX}_{\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))}$, focusing on the transformation and representation in both the tokens heap and base constructs.

Table 5: Structure Components of $\mathbf{SX}_{\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))}$

	Tokens Heap	Base
Initial object	$\varnothing$	$\varnothing$
Terminal object	$\left(\{0\} \coprod E\right)^{\&}$	$\{0\}$
Product	$\left(\left(p \coprod 1_E\right)^{\&}\right)^{-1}\left(\Upsilon_1\right) \cap$ $\left(\left(q \coprod 1_E\right)^{\&}\right)^{-1}\left(\Upsilon_2\right)$	$A \times B$
Coproduct	$\left(\left(inc_1 \coprod 1_E\right)^{\&}\right)^{-1}\left(\Upsilon_1\right) \cup$ $\left(\left(inc_2 \coprod 1_E\right)^{\&}\right)^{-1}\left(\Upsilon_2\right)$	$A \coprod B$
Exponents	$\left(A^B \coprod E\right)^{\&}-$ $\left(p_m\right)^{\&}\left(\left(q_m\right)^{\&}\right)^{-1}\left(\Upsilon_2\right)-$ $\left(evl^{\&}\right)^{-1}\left(\Upsilon_1\right)$	A^B
Limit	$\lim\limits_{j \in J} v_j^{-1}(F(j))$	$\underline{\mathbf{Lim}}\left(P \mathcal{U}^{-1} \mathcal{M}^{-1}(Or \circ F)\right)$
Limiting cone	$in_j \circ v_j \circ$ $\mathbf{in}\left(\underline{\mathbf{Lim}} F, \underline{\mathbf{Lim}} Or \circ F\right)$	-
Truth object	$\left(\{0,1\} \coprod E\right)^{\&}$	$\{0,1\}$

The structure components of $TS_{\mathbf{E}}=\mathbf{SX}_{\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))}$ are summarized in table 5 explained as follows.

Initial object. Because

$$\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$$

is isomorphic to $\mathcal{U}(\mathbf{Set} \times \mathbf{E})$, the structure components of

$$\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$$

are obtained simply by mapping them from $\mathcal{U}(\mathbf{Set} \times \mathbf{E})$. As shown in table 3, initial object of $\mathcal{U}(\mathbf{Set} \times \mathbf{E})$ is $\left(\varnothing \coprod E\right)$, then initial object of

$$\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))$$

is $\left(\varnothing \coprod E\right)^{\&}$. Considering the relation between $\mathbf{Sc}$ and $\mathbf{SX}_{\mathbf{Sc}}$, the initial object of

$$TS_{\mathbf{E}}=\mathbf{SX}_{\mathcal{M}(\mathcal{U}(\mathbf{Set} \times \mathbf{E}))}$$

is the p-class $\langle \varnothing, E, \varnothing\rangle$ denoted as $Ini_{TS_{\mathbf{E}}}$.

Terminal object. Similarly, we have that terminal object is the p-class

$$Ter_{TS_{\mathbf{E}}} = \left\langle \{0\}, E, \left(\{0\} \coprod E\right)^{\&} \right\rangle$$

products. Let $\Upsilon_1 \subset (A \coprod E)^{\&}$ and $\Upsilon_2 \subset (B \coprod E)^{\&}$ for some $A, B \in \mathbf{Set}$, arrows $p : A \times B \to A$ and $q : A \times B \to B$ be projection maps; arrows p and q in diagram

$$A \stackrel{p}{\to} A \times B \stackrel{q}{\leftarrow} B$$

be inclusion maps. Then the product of p-classes $\langle A, E, \Upsilon_1\rangle$ and $\langle B, E, \Upsilon_2\rangle$ is the p-class

$$\left\langle A \times B, E, \left(\left(p \coprod 1_E\right)^{\&}\right)^{-1} (\Upsilon_1) \cap \left(\left(q \coprod 1_E\right)^{\&}\right)^{-1} (\Upsilon_1) \right\rangle.$$

Coproducts. Let arrows inc_1 and inc_2 in diagram

$$A \stackrel{inc_1}{\to} A \coprod B \stackrel{inc_2}{\leftarrow} B$$

be inclusion maps. then the coproduct of p-classes $\langle A, E, \Upsilon_1\rangle$ and $\langle B, E, \Upsilon_2\rangle$ is the p-class

$$\left\langle A \coprod B, E, \left(\left(inc_1 \coprod 1_E\right)^{\&}\right)^{-1} (\Upsilon_1) \cup \left(\left(inc_2 \coprod 1_E\right)^{\&}\right)^{-1} (\Upsilon_1) \right\rangle.$$

Limits. Let P and Q in diagram

$$\mathbf{Set} \stackrel{P}{\leftarrow} \mathbf{Set} \times \mathbf{E} \stackrel{Q}{\to} \mathbf{E}.$$

be product projections of category $\mathbf{Set} \times \mathbf{E}$. Given a finite category J and $F : J \to \mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{Set}\times\mathbf{E})))}$, we have

$$\underline{\mathbf{Lim}} Or \circ F = \left(\left(\underline{\mathbf{Lim}} \left(P\mathcal{U}^{-1}\mathcal{M}^{-1}(Or \circ F)\right)\right) \coprod E\right)^{\&}.$$

Let

$$v : \underline{\mathbf{Lim}} Or \circ F \to Or \circ F$$

be the limiting cone in

$$\mathcal{M}\left(\mathcal{G}\left(\mathcal{U}\left(\mathbf{Set} \times \mathbf{E}\right)\right)\right)$$

and

$$v_j : \underline{\mathbf{Lim}} Or \circ F \to Or \circ F(j).$$

The limit in

$$\mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{Set}\times\mathbf{E})))}$$

is

$$\underline{\mathbf{Lim}}F \cong \left\langle \underline{\mathbf{Lim}} \left(P\mathcal{U}^{-1}\mathcal{M}^{-1}(Or \circ F) \right), E,_{j \in J} v_j^{-1} \left(F(j) \right) \right\rangle .$$

Exponents. Let arrow evl in diagram

$$\left(A^B \coprod E \right)^{\&} \times \left(B \coprod E \right)^{\&} \overset{evl}{\rightarrow} \left(A \coprod E \right)^{\&}$$

(2)

be the evaluation map, then it is constructed from the evaluation map $evl_{\mathbf{Set}}$

$$A^B \times B \overset{evl_{\mathbf{Set}}}{\rightarrow} A,$$

and the evaluation map $evl_{\mathbf{E}} = 1_E$

$$E^E \times E \overset{evl_{\mathbf{E}}}{\rightarrow} E.$$

as

$$evl = \mathcal{M}\left(\mathcal{U}\left(\left(evl_{\mathbf{Set}}, 1_E \right) \right) \right) = \left(evl_{\mathbf{Set}} \coprod 1_E \right)^{\&}$$

where

$$\left(\left(A^B, E \right) \times (B, E) \right) \overset{\left(evl_{\mathbf{Set}}, evl_{\mathbf{E}} \right)}{\rightarrow} (A, E)$$

$$\left(\left(A^B \coprod E \right)^{\&} \times \left(B \coprod E \right) \right)^{\&} \overset{\left(evl_{\mathbf{Set}} \coprod evl_{\mathbf{E}} \right)^{\&}}{\rightarrow} \left(A \coprod E \right)^{\&}$$

and arrows p_m and q_m are the projection maps in

$$\left(A^B \coprod E \right)^{\&} \overset{p_m}{\leftarrow} \left(A^B \coprod E \right)^{\&} \times \left(B \coprod E \right)^{\&} \overset{q_m}{\rightarrow} \left(B \coprod E \right)^{\&}.$$

(3)

Then exponent of p-classes $\langle A, E, \Upsilon_1 \rangle$ and $\langle B, E, \Upsilon_2 \rangle$ is

$$\langle A, E, \Upsilon_1 \rangle^{\langle A, E, \Upsilon_2 \rangle} = \left\langle A^B, E, \Upsilon \right\rangle$$

where

$$\Upsilon = \left(A^B \coprod E \right)^{\&} - (p_m)^{\&} \left(\left((q_m)^{\&} \right)^{-1} (\Upsilon_2) - \left((evl)^{\&} \right)^{-1} (\Upsilon_1) \right).$$

Truth. For any two Token class $T_1 = \langle B_1, E, \Upsilon_1 \rangle$ and $T_2 = \langle B_2, E, \Upsilon_2 \rangle$, if $B_1 \subset B_2$ and $B_1 \subset B_2$, then we call T_1 is a subobject, or a *subclass*, of T_2, and write as $T_1 \Subset T_2$.The truth object is

$$Truth_{TS_{\mathbf{E}}} = \left\langle \{0,1\}, E, \left(\{0,1\} \coprod E\right)^{\&} \right\rangle,$$

though we can not specify a subobject by a single pullback square, Instead, we use two pullback squares.

Definition 2. *Functor* $\circledcirc : TS_{\mathbf{E}} \to TS_{\mathbf{E}}$ *which sends* $T = \langle B, E, \Upsilon \rangle$ *to* $\circledcirc(T) = \langle \Upsilon, E, \varnothing \rangle$ *and sends arrows* $f = \langle f, 1_E, p \rangle : T_1 \to T_2$ *to*

$$\langle p, 1_E, \varnothing \rangle : \circledcirc(T_1) \to \circledcirc(T_2)$$

is called the abstracting *functor. Functor* $\circledast : TS_{\mathbf{E}} \to TS_{\mathbf{E}}$ *defined as*

$$\circledast(\langle B, E, \Upsilon \rangle) = \left\langle B, E, \left(B \coprod \{E\}\right)^{\&} \right\rangle$$

and

$$\circledast(\langle f, 1_E, p \rangle) = \left\langle f, 1_E, \left(f \coprod 1_E\right)^{\&} \right\rangle$$

is called stuffing *functor.*

Theorem 2. *For any two Token class* $T_1 = \langle B_1, E, \Upsilon_1 \rangle$ *and* $T_2 = \langle B_2, E, \Upsilon_2 \rangle$ *and p-map* $\langle f, E, p \rangle : T_1 \to T_2$*. If a pullback square*

$$\begin{array}{ccc} \circledast T_1 & \longrightarrow & Ter_{TS_E} \\ \Big\downarrow{\scriptstyle \circledast f} & & \Big\downarrow{\scriptstyle t} \\ \circledast T_2 & \xrightarrow{\psi} & Truth_{TS_E} \end{array}$$

together with a pullback square

$$\begin{array}{ccc} \circledast(\circledcirc T_1) & \longrightarrow & Ter_{TS_E} \\ \Big\downarrow{\scriptstyle \circledast(\circledcirc f)} & & \Big\downarrow{\scriptstyle t} \\ \circledast(\circledcirc T_2) & \xrightarrow{\varphi} & Truth_{TS_E} \end{array}$$

exist, then T_1 *is a subclass of* T_2*.*

Proof. This is simple. ∎

Because there are category structures to specify subobjects in an element Token space, we also call an element Token space a *Token topos.*

4.1 Token Space

Let

$$U =_{A \in \mathbf{Set}} A$$

We assume that $U \subset \mathbf{Set}$. The elementary Token space

$$TS = \mathbf{SX}_{\mathcal{M}(\mathcal{U}(-))}(\mathbf{Set} \times \mathbf{U})$$

is also called *Token space*. Each object $\langle B, U, \Upsilon\rangle$ in TS is also represented as

$$\langle B, E, \Upsilon\rangle$$

for any E such that $\Upsilon \subset \mathcal{M}(\mathcal{U}\langle B, E\rangle)$. Thus $\langle B, E, \Upsilon\rangle$ is viewed isomorphic to $\langle B, U, (I \coprod 1_B)(\Upsilon)\rangle$, where $I : E \to U$ send each x in E to x in U, in TS. If there is a p-map from $\langle B_1, E_1, \Upsilon_1\rangle$ to $\langle B_2, E_2, \Upsilon_2\rangle$ in TS, then there must be $B_1 \subset B_2$ and $\Upsilon_1 \subset \Upsilon_2$; not necessarily $E_1 \subset E_2$.

Remark 1. *Is* $_{A\in\mathbf{Set}}A$ *a set? We can not know all sets in* **Set**. *Evidently, the union of all sets computed from existing sets is a set. We use* U *to denote such a set. Neverthless we know only part of* U *at any time.*

In a category, two object a and b are isomorphic if there is an invertible arrow, i.e. an isomorphism, from a to b. For any p-classes T_1, T_2, T_3, there are isomorphisms:

$$T_1 \otimes T_2 \cong T_2 \otimes T_1$$

$$T_1 \oplus T_2 \cong T_2 \oplus T_1$$

$$(T_1 \otimes T_2) \otimes T_3 \cong T_1 \otimes (T_2 \otimes T_3)$$

$$(T_1 \oplus T_2) \oplus T_3 \cong T_1 \oplus (T_2 \oplus T_3)\,.$$

5 Representing Categories of Structured Objects in Token Space

How structures and structure preserving map are defined in mathematics? The general way is:

- There are some sets, say, $A, B, C \ldots$, which are under consideration.
- In each set of them, there are some special element elements. Structure persevering map needs to map these elements correspondingly. For example, in $\mathbf{Set}_{\&}$ the only special elements of a set A is the base points $\&_A$, and the structure preserving from A to B should map $\&_A$ to $\&_B$. In algebras, the special elements are called nullary operations.
- In each set of them, There are some relations or operations on it. Operations can be regarded as relations which connect elements that operators applied on and the resulting elements. For example, $+$ is a relation which connects x, y to z if $x + y = z$. Structure preserving map should preserve these relations such that the same connections between elements exist when they are mapped to another set.
- There are some properties which each set and its special elements in it, relations and operations on it should satisfy. **However, such properties are always irrelevant to the structure preserving map.** For example, a semigroup $\mathbf{Sg} = \langle Sg, \cdot\rangle$ with $\cdot$ a binary operator, and satisfies identity

  $$x \cdot (y \cdot z) = (x \cdot y) \cdot z$$

  (4)

But the definition of homomorphism between semigroups is not related to this identity which states the associativities of operator $\cdot$.

Definition 3. *A* structure *on a set A is a binary $\langle\Theta,\Gamma\rangle$ with Θ the set of symbols representing the special elements in A, Γ the set of symbols, each of which has a fixed arty, representing relations or operators on A. A* structure preserving map *from A to B under Θ and Γ is a map $f:A\to B$ such that f map elements in Θ to itself and any tuple $r=(x_1,x_2,\cdots,x_n)$, if $r\in R$ for some $R\in\Gamma$, then $(f(x_1),f(x_2),\cdots,f(x_n))\in R$. A category $\mathbf{C}$ of structured objects under Θ and Γ is a category of sets with structure $\langle\Theta,\Gamma\rangle$ and structure preserving maps between them.*

Given a category $\mathbf{C}$ of structured objects under Θ and Γ, we can define some functors from $\mathbf{C}$ to TS such that they can be mapped to a isomorphic subcategory of TS whereby we can discuss the categorical structure of $\mathbf{C}$ in TS. Given a category $\mathbf{C}$ of structured objects under (Θ,Γ), we define a functor $\mathcal{R}ep_{\mathbf{C}}:\mathbf{C}\to TS$ as follows:

1.
$$\mathcal{R}ep_{\mathbf{C}}(A)=\langle A,\Theta\cup\Gamma,\Upsilon\rangle$$
where
$$\Upsilon=A\cup\{(\theta,x)\,|\theta_A=x,\theta\in\Theta\ \}\cup$$
$$\{(\gamma,x_1,x_2,\cdots,x_n)\,|(x_1,x_2,\cdots,x_n)\in\gamma_A,n\in\mathbb{N},\gamma\in\Gamma\ \}$$

2. if $f:A\to B$ is a structure preserving map from $\langle A,\Theta,\Gamma\rangle$ to $\langle B,\Theta,\Gamma\rangle$ under Θ and Γ, then $\mathcal{R}ep_{\mathbf{C}}(f)$ is the p-map
$$\left\langle f,1_{\Theta\cup\Gamma},\mathbf{on}\left(\mathcal{M}\left(\left(f\coprod 1_{\Theta\cup\Gamma}\right)^{\&}\right),\Upsilon_{\mathcal{R}ep_{\mathbf{C}}(A)},\Upsilon_{\mathcal{R}ep_{\mathbf{C}}(B)}\right)\right\rangle$$
from $\mathcal{R}ep_{\mathbf{C}}(A)$ to $\mathcal{R}ep_{\mathbf{C}}(B)$.

Now we investigate how mathematics objects in many categories can be regarded as structured objects.

Category Set of sets. Each object A in **Set** has structure $\langle\varnothing,\varnothing\rangle$, then
$$\mathcal{R}ep_{\mathbf{Set}}(A)=\langle A,\varnothing,\varnothing\rangle$$
for each arrow (a function) $f:A\to B$ in **Set**, and
$$\mathcal{R}ep_{\mathbf{Set}}(f)=\langle f,\varnothing,\varnothing\rangle\,.$$
is a structure preserving map.

Category $\mathbf{Set}_{\&}$ of pointed sets. Each object A with a base point & in $\mathbf{Set}_{\&}$ has structure $\langle\{\&\},\varnothing\rangle$, then
$$\mathcal{R}ep_{\mathbf{Set}_{\&}}(A)=\langle A,\{\&\},\{(\&,\&_A)\}\rangle$$
and each arrow $f:A\to B$ (a function mapping base point to base point) in $\mathbf{Set}_{\&}$ can be represented as

$$\mathcal{R}ep_{\mathbf{Set}_{\&}}(f) = \langle f, \{\& \mapsto \&\}, \{(\&, \&_A) \mapsto (\&, \&_B)\}\rangle \,.$$

Category Ords of ordered sets. Each object $\mathbf{P} = \langle P, \leqslant\rangle$ in category of ordered set **Ords** has structure $\langle \varnothing, \{\leqslant\}\rangle$, then

$$\mathcal{R}ep_{\mathbf{Ords}}(\mathbf{P}) = \langle P, \{\leqslant\}, \{(\leqslant, x, y)\,|x \leqslant y\ \}\rangle$$

and each arrow $f : \mathbf{P}_1 \to \mathbf{P}_2$ (a homomorphisms between two ordered sets) in **Ords** can be represented as

$$\mathcal{R}ep_{\mathbf{Ords}}(f) = \langle f, \{\leqslant \mapsto \leqslant\}, \{(\leqslant, x, y) \mapsto (\leqslant, f(x), f(y))\,|x \leqslant y\ \}\rangle \,.$$

Category Gph of graphs. Each object $\mathbf{G} = \langle V, Edg\rangle$ in category of graph **Gph** has structure $\langle \varnothing, \{Edg\}\rangle$, then

$$\mathcal{R}ep_{\mathbf{Gph}}(\mathbf{G}) = \langle V, \{Edg\}, \{(Edg, x, y)\,|(x, y) \in Edg\ \}\rangle$$

and each arrow $f : \mathbf{G}_1 \to \mathbf{G}_2$ (a homomorphisms between two graphs) in **Gph** can be represented as

$$\mathcal{R}ep_{\mathbf{Gph}}(f) = \langle f, \{Edg \mapsto Edg\}, \{(Edg, x, y) \mapsto (Edg, f(x), f(y))\,|(x, y) \in Edg\ \}\rangle \,.$$

Category Rng of rings. Each object $\mathbf{R} = \langle R, +, \cdot, -, 0\rangle$ in category of ring **Rng** has structure $\langle \{0\}, \{+, \cdot, -\}\rangle$, then

$$\mathcal{R}ep_{\mathbf{Rng}}(\mathbf{R}) = \langle R, E, \Upsilon\rangle$$

where

$$E = \{+, \cdot, -, 0\}$$

and

$$\Upsilon = \{(+, x, y, z)\,|x + y = z\ \} \cup$$

$$\{(\cdot, x, y, z)\,|x \cdot y = z\ \} \cup$$

$$\{(-, x, y)\,|-x = y\ \} \cup$$

$$\{(0, 0_{\mathbf{R}})\}$$

and each arrow $f : \mathbf{R}_1 \to \mathbf{R}_2$ (a homomorphisms between two rings) in **Rng** can be represented as

$$\mathcal{R}ep_{\mathbf{Rng}}(f) = \langle f, 1_E, p\rangle$$

where

$$p = \{(+, x, y, z) \mapsto (+, f(x), f(y), f(z))\,|x + y = z\ \} \cup$$

$$\{(\cdot, x, y, z) \mapsto (\cdot, f(x), f(y), f(z))\,|x \cdot y = z\ \} \cup$$

$$\{(-, x, y) \mapsto (-, f(x), f(y))\,|-x = y\ \} \cup$$

$$\{(0, 0_{\mathbf{R}_1}) \mapsto (0, 0_{\mathbf{R}_2})\}$$

Category Vct of vector spaces. Each object $\mathbf{V} = \langle V, 0, +, \mathbb{R}\rangle$ in category of vector space **Vct** has structure $\langle\{0\}, \{+\} \cup \mathbb{R}\rangle$, then

$$\mathcal{R}ep_{\mathbf{Vct}}\left(\mathbf{V}\right) = \langle V, E, \Upsilon\rangle$$

where

$$E = \{0, +\} \cup \mathbb{R}$$

and

$$\Upsilon = \{(+, x, y, z)\,|x + y = z\ \} \cup$$

$$\{(a, x, y)\,|ax = y, a\in \mathbb{R}\ \} \cup$$

$$\{(0, 0_{\mathbf{V}})\}$$

and each arrow $f : \mathbf{V}_1 \to \mathbf{V}_2$ (a linear transformation between two vector spaces) in **Vct** can be represented as

$$\mathcal{R}ep_{\mathbf{Vct}}(f) = \langle f, 1_E, p\rangle$$

where

$$p = \{(+, x, y, z) \mapsto (+, f(x), f(y), f(z))\,|x + y = z\ \} \cup$$

$$\{(a, x, y) \mapsto (a, f(x), f(y))\,|ax = y\ \} \cup$$

$$\{(0, 0_{\mathbf{V}_1}) \mapsto (0, 0_{\mathbf{V}_2})\}$$

Because $\mathbb{R}$ in $\mathbf{V}$ is regarded as a ring $\mathbf{R} = \langle \mathbb{R}, +_{\mathbb{R}}, \cdot_{\mathbb{R}}, -_{\mathbb{R}}, 0_{\mathbb{R}} \rangle$, a vector space $\mathbf{V}$ can be viewed as set $V \cup \mathbb{R}$ with structure $\langle \{0, 0_{\mathbb{R}}\}, \{+, +_{\mathbb{R}}, \cdot_{\mathbb{R}}, -_{\mathbb{R}}\} \cup \{V, \mathbb{R}\} \rangle$, and

$$\mathcal{R}ep'_{\mathbf{Vct}}(\mathbf{V}) = \left\langle V \cup \mathbb{R}, E', \Upsilon' \right\rangle$$

where

$$E' = \{0_V, 0_{\mathbb{R}}, +_V, +_{\mathbb{R}}, \cdot_{\mathbb{R}}, -_{\mathbb{R}}, V, \mathbb{R}\}$$

and

$$\Upsilon' = \{(+, x, y, z) \,|x + y = z, x, y, z \in V \ \} \cup$$

$$\{(a, x, y) \,|ax = y, x, y \in V, a \in \mathbb{R} \ \} \cup$$

$$\{(+_{\mathbb{R}}, a, b, c) \,|a + b = c, a, b, c \in \mathbb{R} \ \} \cup$$

$$\{(\cdot_{\mathbb{R}}, a, b, c) \,|a \cdot b = c, a, b, c \in \mathbb{R} \ \} \cup$$

$$\{(-_{\mathbb{R}}, a, b) \,|-a = b, a, b \in \mathbb{R} \ \} \cup$$

$$\{(0, 0_{\mathbf{V}}), (0_{\mathbb{R}}, 0_{\mathbf{R}})\} \cup$$

$$\{(V, x) \,|x \in V \ \} \cup$$

$$\{(\mathbb{R}, a) \,|a \in \mathbb{R} \ \}$$

Each linear transformation $f : \mathbf{V}_1 \to \mathbf{V}_2$ can be represented as usual.

Opposite category $\mathbf{Top}^{op}$ of topological spaces. All the objects just mentioned can be easily represented as structured objects. However, for the structure on a topological space, there need a spacial treatment. Because the continues map is defined as that the original image of an open set is still an open set, we can hardly define the continues map derectly as structued preserving map. Instead, the reverse of a continues map can be regard as a structure preserving map directly. Each object $\mathbf{X}$ in opposite category of topological space $\mathbf{Top}^{op}$ can be viewed as an object $\{e\} \cup \mathbf{X} \cup \mathcal{P}(\mathbf{X})$ of structure $\langle \{\varepsilon\}, \{\gamma_{\mathcal{P}}, \gamma_{\mathcal{O}}\} \rangle$, where $e \notin \mathbf{X} \cup \mathcal{P}(\mathbf{X})$,

$$\gamma_{\mathcal{P}} = \{(S, x) \,|S \in \mathcal{P}(\mathbf{X}), x \in S \ or \ x = e \ \}$$

and

$$\gamma_{\mathcal{O}} = \{S \,|S \ is \ an \ open \ set \ on \ \mathbf{X} \ \}.$$

If $f:\mathbf{X}\to\mathbf{Y}$ is a continuous map, f^{-1} does not defined on $\mathbf{Y}$ but $\mathcal{P}(\mathbf{Y})$. Given a continuous map $f:\mathbf{X}\to\mathbf{Y}$, we define

$$f^{op}:\{e\}\cup\mathbf{Y}\cup\mathcal{P}(\mathbf{Y})\to\{e\}\cup\mathbf{X}\cup\mathcal{P}(\mathbf{X})$$

as

$$f^{op}(y)=\begin{cases}f^{-1}(y), y\in\mathcal{P}(\mathbf{Y})\\ e, y\notin\mathcal{P}(\mathbf{Y})\end{cases}.$$

Proposition 8. *Given two continuous map $f,g:\mathbf{X}\to\mathbf{Y}$, if $f\neq g$ then $f^{op}\neq g^{op}$.*

Proof. This is simple. □

Proposition 9. *Given a continuous map $f:\mathbf{X}\to\mathbf{Y}$, f^{op} is a structure ipreserving map from $\{e\}\cup\mathbf{Y}\cup\mathcal{P}(\mathbf{Y})$ to $\{e\}\cup\mathbf{X}\cup\mathcal{P}(\mathbf{X})$.*

Proof. This is simple. □

Definition 4. Powering ∇*: Let $\gamma,\varepsilon,e\notin B\cup E$, $\mathcal{P}(B)$ be the power set of B, then*

$$\nabla(\langle B,E,\Upsilon\rangle)=\langle\{e\}\cup B\cup\mathcal{P}(B),E\cup\{\varepsilon,\gamma\},\Upsilon_T\rangle$$

where

$$\Upsilon_T=\{(\gamma,S,x)\,|S\in\mathcal{P}(B),x\in S or x=e\ \}\cup$$

$$\left\{(x_1,x_2,\cdots,x_n)\,\middle|\left(x_1',x_2',\cdots,x_n'\right)\in\Upsilon,x_i'\in x_i or x_i'=x_i\ \right\}$$

and

$$\nabla(\langle f,1_E,p\rangle)=\left\langle f\cup\{e\mapsto e\}\cup\mathcal{P}(f),1_E\cup\{\varepsilon\mapsto\varepsilon,\gamma\mapsto\gamma\},p'\right\rangle$$

where

$$p'=p\cup\mathbf{on}\left(\mathcal{M}\left((f\cup\{e\mapsto e\}\cup\mathcal{P}(f))^{\&}\right),\Upsilon_{\nabla T_1},\Upsilon_{\nabla T_2}\right).$$

forgetting $\varnothing$ *defined as*

$$\varnothing T=\langle\mathbf{base}(T),E,\varnothing\rangle,$$

and

$$\varnothing(\langle f,1_E,p\rangle)=\langle f,1_E,\varnothing\rangle;$$

Given two p-classes $T_1 = \langle B_1, E, \Upsilon_1\rangle$, $T_2 = \langle B_2, E, \Upsilon_2\rangle$ and p-map $f : T_1 \to T_2$, Reversing $\rightleftharpoons (f)$ is a p-map from $\nabla\varnothing T_2$ to $\nabla\varnothing T_1$ such that

$$\rightleftharpoons (f)(y) = \begin{cases} f^{-1}(y), y \in \mathcal{P}(B_2) \\ e, y \notin \mathcal{P}(B_2) \end{cases}.$$

Therefore, each topological space $\mathbf{X}$ can be represented as a p-class

$$\mathcal{R}ep_{\mathbf{Top}^{op}}(\mathbf{X}) = \langle B, E, \Upsilon \cup \{(\gamma_{\mathcal{O}}, S) \,|S \quad isanopensetonX\ \}\rangle$$

with $\langle B, E, \Upsilon\rangle = \nabla\langle \mathbf{X}, \varnothing, \varnothing\rangle$. The structure preserving map $f : \mathbf{X} \to \mathbf{Y}$ is just is the p-map

$$\rightleftharpoons (f) \colon \mathcal{R}ep_{\mathbf{Top}^{op}}(\mathbf{Y}) \to \mathcal{R}ep_{\mathbf{Top}^{op}}(\mathbf{X}).$$

6 Token Categories

By discussion in above two sections, there are two operators on products of subcategories of **Set**: operator $\mathcal{U}$ and $\mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(-)))}$ – they can form new bi-cartsian closed categories from bi-cartsian closed categories, form categories having all finite limits from categories having all finite limits. Algebra **APC** is a $\{\times, \mathcal{U}, \mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(-)))}\}$-type algebra generated on

$$\{\mathbf{Set}\} \cup \mathbf{idset}(ISets).$$

Each element in **APC** is a category. Below are some examples (where $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are some identity set categories):

$$\mathbf{A}, \mathbf{Set}, \mathbf{A}, \mathbf{Set} \times \mathbf{B}, \mathcal{U}(\mathbf{Set} \times \mathbf{A} \times \mathbf{B} \times \mathbf{C}),$$

$$\mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{A})))}, \mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{Set}\times\mathbf{A})))}, \mathcal{U}\left(\mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{Set}\times\mathbf{A})))}\right),$$

$$\mathcal{U}\left(\mathbf{Set} \times \mathbf{C} \times \mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(-)))}\left(\mathbf{Set} \times \mathbf{B} \times \mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{Set}\times\mathbf{A})))}\right)\right)$$

For any categories **PC** in **APC**, category

$$\mathbf{Ptn}^{\&}_{\mathbf{PC}} = \mathcal{U}(\mathbf{PC})$$

and

$$\mathbf{Ptn}_{\mathbf{PC}} = \mathbf{SX}_{\mathcal{M}(\mathcal{G}(\mathcal{U}(\mathbf{PC})))}$$

are called a *Token category.* Simply, any Token categories are elements in **APC**.

Theorem 3. *Every Token category is a subcategory of* **Set***, is bicartisain closed and has all finite limits.*

Proof. This is simple. □

7 Interior Structure Mapping and Tree Token Classes

7.1 Generation of Tree Tokens

A tree Token on set B is a tuple whose elements may be also tree Tokens or elements in B. For example, $(x_1, (x_2, (x_3, x_4)), x_5)$ with $x_1, \cdots, x_5 \in B$ is a tree Token. This definition is more general than tree formal language [1], because the tree language is defined on ranked symbols of which each has a rank or arity like predicates in logic. Symbols in a tree Token have no rank. For the simplicity of process the tree Token formation and matching, we define tree Tokens in an algebraic way instead of an inductive way.

Definition 5 (Pre-treepoid)**.** *To represent interior stucture, we introduce unary operator & in addition to monoid structure. However the definition is arduous. The pre-*treepoid *$\mathcal{G}(B)$ freely generated on B is an algebra $\langle M, \cdot, \&, \varepsilon\rangle$ with $\cdot$ associative binary operator and unary operator and ε nulary operator. Without , a $\langle \cdot, \varepsilon\rangle$-type algebra is just a monoid. Each r in $\mathcal{G}(B)$ is called a tree Token. The* length *of element $r \in \mathcal{G}(B)$, denoted as $|r|$, and* depth *of r, denoted as $\|r\|$, and tree-arity of r, denoted as $\mathbf{ary}(r)$, are defined inductively as:*

1. $|\varepsilon| = 0$, $\|\varepsilon\| = 0$ *and* $\mathbf{ary}(\varepsilon) = 0$*;*
2. *if* $x \in B$*, then* $|x| = 1$, $\|x\| = 1$ *and* $\mathbf{ary}(x) = 1$*;*
3. $|\&(r)| = max(|r|, 1)$, $\|\&(r)\| = \|r\| + 1$ *and* $\mathbf{ary}(\&(r)) = 1$*;*
4. $|r_1 \cdot r_2| = |r_1| + |r_2|$, $\|r_1 \cdot r_2\| = max(\|r_1\|, \|r_2\|)$ *and* $\mathbf{ary}(r_1 \cdot r_2) = \mathbf{ary}(r_1) + \mathbf{ary}(r_2)$*.*

For any function $f_0 : B_1 \to \mathcal{G}(B_2)$ their is a unique homomorphism $f: \mathcal{G}(B_1) \to \mathcal{G}(B_2)$ such that f is extended from f_0, and f_0 and f uniquely determine each other.

If $f : A \to B$ be a homomorphism, then $\mathbf{ary}(r)$ may not equal to $\mathbf{ary}(f(r))$ for some $r \in A$.

Proposition 10. *For any $r \in \mathcal{G}(B)$, there is a $r' \in \mathcal{G}(B)$ such that $r = r'$, iff $\mathbf{ary}(r) = 1$ and $\|r\| \geq 2$.*

Proof. Clear. □

Definition 6 (Treepoid)**.** *Given a set B, the set*

$\mathcal{T}(B) = B^{\pitchfork} = \{r \,|\mathbf{ary}(r) \leq 1, r \in \mathcal{G}(B)\}$

is called a free tree Algebra on B. Any element in $\mathcal{T}(B)$ is called a treepoid*. Given tree Token r , the arity of r, denoted as $\mathbf{tary}(r)$, is defined as that if $r = \left(r'\right)$ for some $r' \in \mathcal{T}(B)$, then $\mathbf{tary}(r) = \mathbf{ary}\left(r'\right)$; if $r \in B$ then $\mathbf{tary}(r) = 0$. In this case, we denote $r' = \&^{-1}(r)$. Given a set B and an function $f_0 : B_1 \to \mathcal{T}(B_2)$, ther is a unique homomorphism $g : \mathcal{G}(B_1) \to \mathcal{G}(B_2)$ extended from f_0, then we say the set*

$$f = \{r \mapsto g(r) \,|r \in \mathcal{T}(B_1)\}$$

a tree morphism *between $\mathcal{T}(B_1) \to \mathcal{T}(B_2)$ extending f_0, and denoted f as $\mathbf{tmorph}(f_0)$.*

By the definition, it follows that:

1. let $f : B_1^{\pitchfork} \to B_2^{\pitchfork}$ be a tree morphism, then $\mathbf{tary}(r) = \mathbf{tary}(f(r))$ holds for all $r \in B_1^{\pitchfork}$; if $r \neq \varepsilon$, then $f(r) \neq \varepsilon$;
2. $\varepsilon \in B^{\pitchfork}$ and $(\varepsilon) \in B^{\pitchfork}$;
3. $B \subset B^{\pitchfork}$.

We may call the tree monoid the algebra of nested parentheses. We can represent tree Tokens as tuples, similar to representing Tokens in free monoids as tuples:

- Given Token r with

$$\&^{-1}(r) = x_1 \cdot x_2 \cdot \dots \cdot x_n$$

  with $\textbf{tary}\,(x_i) = 1$ for $i = 1, 2, \cdots, n$, i.e. the arity of r is n, and r is not in another Token, then we write $\&(r)$ as

$$(x_1, x_2, \cdots, x_n)\,.$$

  Note in a Token class T, a Token $(x_1, x_2, \cdots, x_n)$ really refers to $x_1 \cdot x_2 \cdot \dots \cdot x_n$. However, a tree Token $(x_1, x_2, \cdots, x_n)$ just refers to $\& \left(x_1 \cdot x_2 \cdot \dots \cdot x_n\right)$. In following sections, we only use $x_1 \cdot x_2 \cdot \dots \cdot x_n$ to denote Token of arity n.

- And if, for example,

$$x_2 = (y_1, y_2, \cdots, y_m)$$

  then r will be represented as

$$(x_1, (y_1, y_2, \cdots, y_m)\,, \cdots, x_n)$$

- For any Token $r \notin B$ we do not use (r) to denote $\&(r)$. Note we will use (ε) to represents $\&(\varepsilon)$.

Using inductive definition, the tree Token can be directly defined as tuple on tuples. This seems more intuitive. However, the inductive defintion of tree monoid indeed use infinite n-tuple elements formation operators $(,,\cdots,)$ for each $n \geq 1$. The algebraic definition of tree monoids can let us use very few operators, i.e., three operators, to form tree structures.

Note for any tree morphism $f\colon \mathcal{T}(B_1) \to \mathcal{T}(B_2)$ and any $x \in B_1$, f can not send x to a Token $x_1 \cdot x_2 \cdot \dots \cdot x_n$ with $n > 1$, but can send x to $(x_1, x_2, \cdots, x_n)$ or $((x_1, x_2, \cdots, x_n))$, etc, because the arities of these Tokens are all equal to 1.

Definition 7. *The* connection operator $\bowtie$ *is used on any two Tokens* $r_1, r_2 \in \mathcal{T}(B)$ *as*

$$r_1 \bowtie r_2 = \left(\&^{-1}(r_1) \cdot \&^{-1}(r_2)\right).$$

We also use connection operator $\bowtie$ on Token sets. Binary $\bowtie$ is associative. The operator $\bowtie$ is also used as unary operator, called flattening*, on Tokens, and is defined as*

$$\bowtie (r) = \bowtie ((x_1, x_2, \cdots, x_n))$$

$$= (\bowtie (x_1)) \bowtie (\bowtie (x_2)) \bowtie \cdots \bowtie (\bowtie (x_n))$$

with the addition of that $\|x\| = 2$ implies

$$\bowtie (x) = x$$

and $\|x\| = 1$, *i.e.* $x \in B$, *implies*

$$\bowtie (x) = (x)$$

Proposition 11. *For any Token* $r \in \mathcal{T}(B)$, *we have*

$$\| \bowtie (r) \| = 2$$

.

Proof. Clear. □

Let Υ, Υ_1 and Υ_2 be sets of tree Tokens, then

$$\Upsilon_1 \bowtie \Upsilon_2 = \{ r_1 \bowtie r_2 \, | r_1 \in \Upsilon_1, r_2 \in \Upsilon_2, m, n \in \mathbb{N} \ \} .$$

and

$$\bowtie \Upsilon = \{ \bowtie r \, | r \in \Upsilon, n \in \mathbb{N} \ \} .$$

Flattening $\bowtie$ can *condense* a Token:

$$\bowtie (x_1, (\varepsilon), x_3, \cdots, x_n) = (x_1, x_3, \cdots, x_n)$$

The length of $(x_1, (\varepsilon), x_3, \cdots, x_n)$ is n, but that of $\bowtie (x_1, (\varepsilon), x_3, \cdots, x_n)$ is $n - 1$. Therefore, unary $\bowtie$ can not act as a functor. There is a tree monoid homomorphism $f : A \to B$ does not imply that there exists a homomorphism $g : \bowtie (A) \to \bowtie (B)$.

Proposition 12. *Relation* $Eq_{\bowtie}$ *defined as that* $Eq_{\bowtie}(x, y)$ *if and only if* $\bowtie x = \bowtie y$ *is a equivalent relation. We use* $x / \bowtie$ *to denote the equivalent class* $x / Eq_{\bowtie}$, *then*

$$x / \bowtie = \bowtie^{-1} (\bowtie (x)) .$$

Proof. This is simple. □

Definition 8. *Let Token* $r \in \mathcal{T}(B_1)$ *and* $S \subset A$ *just contain all elements appear in* r, *and* $f \colon \mathcal{T}(B_1) \to \mathcal{T}(B_2)$ *be a tree morphism, then the set*

$$g = \{ x \mapsto f(x) \, | x \in S \ \}$$

is called a tree correspondence *between* r *and* $f(r)$.

Theorem 4 (uniqueness of tree correspondence)**.** *From a Token* r_1 *to* r_2, *there is at most one tree correspondence, denoted as* $\mathbf{corr}\,(r_1, r_2)$ *if exists, between* r_1 *and* r_2.

Proof. Clear. □

Definition 9 (universal Token)**.** *We use* $1:n$ *to denote the Token*

$$(1,2,3,\cdots,n)$$

and

$$1:0=\varepsilon$$

$$1:1=(1)$$

Each Token in

$$\bowtie^{-1}\{1:n\,|n\in\mathbb{N}\ \}.$$

is called a universal Token.

Theorem 5 (universal Token of a tree Token)**.** *For any Token* r*, there is a unique universal Token* u *such that a tree correspondence between* u *and* r *exists and*

$$|u|=|r|.$$

We denote such u *as* **universal**(r)*.*

Proof. Clear. ☐

Corollary 5. *For any Token* r*, if* $|r|=n$*, then* $|\bowtie(r)|=n$ *and*

$$\mathbf{universal}\,(\bowtie(r))=1:n.$$

Definition 10 (Scattering)**.** *Let* $r\in\mathcal{T}(B)$ *and a tree correspondence* g *from a universal Token* u *of length* $m\geq1$ *to* r *exist. Then the scattering of* r *by* g *is*

$$\ll_g(r)=\{g(i)\,|i\in\{1,2,\cdots,m\}\ \}.$$

The flat scattering of r *is defined as*

$$\ll(r)=\ll_{\mathbf{corr}(1:\|r\|,\bowtie(r))}(\bowtie(r))$$

Let S *be the set of universal Tokens of which there is a nontrivial tree correspondence from each to* r*, the star scattering of* r *is defined as*

$$\ll^*(r)=_{u\in S}\ll_{\mathbf{corr}(u,r)}(r).$$

Proposition 13. *It follows that (1)* $\ll(r)\subset\ll^*(r)$*; (2)* $r\in\ll^*(r)$*; (3) For any* $r'\in\ll^*(r)$*, we have* $\left\|r'\right\|\leq\|r\|$*;*

Proof. Clear. ☐

7.2 Tokens Maps between Tree Token Classes

To represent interior structure mapping, we need the structure is represented directly. Otherwise, we need parse interior structure from strings. $(x_1, (x_2, (x_3, x_4)), x_5)$. **Note we will use $\mathcal{T}(A)$ itself to denote the underlying set of $\mathcal{T}(A)$ in following**.

Definition 11 (Tree Token Class)**.** *A* tree Token class*, or a* t-class*, is a ternary $T = \langle B, E, \Upsilon\rangle$ where $\Upsilon \in \mathcal{T}(B \coprod E)$, and B, E and Υ, which are denoted as* **base**(T)*,* **core**(T) *and* **heap**(T) *respectively, are called the* base*,* core *and* Token heap *of T; tuples in Υ are called* tree Tokens*. A t-class T_1 is a* subclass *of T_2, denoted as $T_1 \Subset T_2$, iff* **base**$(T_1) \subset$ **base**(T_2) *and* **heap**$(T_1) \subset$ **heap**(T_2)*.*

Definition 12 (Tree Tokens Map)**.** *Given two t-classes $T_1 \langle B_1, E_1, \Upsilon_1\rangle$ and $T_2 = \langle B_2, E_2, \Upsilon_2\rangle$ of same structure E, a* tree Tokens map*, or* t-map*, from T_1 to T_2 extending function*

$$f_0 : B_1 \to \mathcal{T}\left(B_2 \coprod E\right) - E$$

is a ternary $f = \langle f_0, 1_E, p\rangle$ such that

$$p(r) = \mathbf{tmorph}(f)(r)$$

where **tmorph**(f) *is to denote* **tmorph**$(f_0 \coprod 1_E)$*, for all $r \in$* **heap**(T_1)*. And we use* $\mathbf{tmap}_{T_1}^{T_2}(f_0)$*, or*

$$\mathbf{tmap}_{T_1}^{T_2}\left(\mathbf{tmorph}\left(f_0 \coprod 1_E\right)\right)$$

to denotes f. If $\left\langle B_1', E, \Upsilon_1'\right\rangle \Subset \langle B_1, E, \Upsilon_1\rangle$, then there is a t-map f' extends

$$\{x \mapsto f_0(x) \,\Big|\, x \in B_1' \}$$

and we call f' is a submap *of f. If t-map f is extended from f_0, we also called f is a submap of* **tmorph**(f)*, and any submap of f is also a sub map of* **tmorph**(f)*.*

Note **tmorph**$(f_0 \coprod 1_E)$, i.e., **tmorph**(f), is a proper homomorphism. Note for the simplicity, given a t-class $\langle B, E, \Upsilon\rangle$, we always assume that

$$B \cap E = \varnothing$$

and

$$B \coprod E = B \cup E.$$

However, unlike the definition in Token space, a tree Tokens map f from $\langle B_1, E, \Upsilon_1\rangle$ to $\langle B_2, E, \Upsilon_2\rangle$ can map $x \in B_1$ to $f(x) \neq B_2$.

Definition 13 (Metavocabulary)**.** *There is a set* **terms** *called the set of meta-vocabulary, it contains all operators we will used to operate on t-classes. For example, $\cdot, \&, \circledast \in$* **terms***.*

Definition 14 (Tree Token Space)**.** *Let*

$$U =_{A \in \mathbf{Set}} A$$

We assume that $U \subset$ **Set**. *All tree Token classes with core* U *and Tokens maps among them constitute a category* tree Token space TTS. *When we say any t-class* $\langle B, E, \Upsilon \rangle \in TTS$, *then* $\langle B, E, \Upsilon \rangle$ *is viewed isomorphic to* $\langle B, U, (I \coprod 1_B)(\Upsilon) \rangle$, *where* $I : E \to U$ *send each* x *in* E *to* x *in* U. *The set* $\hom_{TTS}(T_1, T_2)$, *or* $TTS(T_1, T_2)$, *denotes the set of all tree Tokens maps between* T_1 *and* T_2.

In following sections we assume all t-classes are in TTS, thus for given B and Υ, t-classes $\langle B, E, \Upsilon \rangle$ and $\langle B, U, \Upsilon \rangle$ are viewed as the same t-class. Therefore the t-map

$$f : \langle B_1, E_1, \Upsilon_1 \rangle \to \langle B_2, E_2, \Upsilon_2 \rangle$$

extending $f_0 : B_1 \to B_2$ is refered to the t-map

$$f : \langle B_1, U, \Upsilon_1 \rangle \to \langle B_2, U, \Upsilon_2 \rangle$$

extending $f_0 : B_1 \to B_2$.

The relations between TTS and TS are specified as follows.

Proposition 14. *Any two t-classes* $Tr_1 = \langle B_1, E, \Upsilon_1 \rangle$ *and* $Tr_2 = \langle B_2, E, \Upsilon_2 \rangle$ *have a coproduct* $Tr_1 \coprod Tr_2$ *in* TTS, *which is isomorphism to*

$$\left\langle B_1 \coprod B_2, E, \mathcal{T}\left(inc_1 \coprod 1_E\right)(\Upsilon_1) \cup \mathcal{T}\left(inc_2 \coprod 1_E\right)(\Upsilon_2) \right\rangle$$

where $inc_1 : B_1 \to B_1 \coprod B_2$ *and* $inc_2 : B_2 \to B_1 \coprod B_2$ *are the inclusion map.*

Proof. Clear. □

Remark 2 (Convention for Using Tree Tokens as t-classes)**.** *For a tree Token set* Υ, *the* base *of* Υ *is the minimum set* B *such that* $\Upsilon \subset \mathcal{T}(B \coprod E)$. *We use* $\mathbf{base}(\Upsilon)$ *to denote such* B. *Notice that* **base** *is an overloaded term since we have used* $\mathbf{base}(T)$ *to denote the base of a p-class or tree Token class* T. *We may use sets of Tokens as t-classes in following sections. A Tokens set* Υ *will be used as a t-class*

$$\langle \mathbf{base}(\Upsilon), E, \Upsilon \rangle .$$

Note

$$\mathbf{base}(\{r\}) = \ll^* (r) \cap B.$$

Remark 3 (Proper tree homomorphism applyed on t-classes)**.** *A proper tree homomorphism* f *from* $\mathcal{T}(B \coprod E)$ *can be applied on any t-class* T *with base* B *and structure* E *as a t-map*

$$\langle \{x \mapsto f(x) \,|x \in B \ \}, 1_E, \{r \mapsto f(r) \,|r \in \mathbf{heap}(T) \ \} \rangle .$$

8 Exploring Structure Relations of Token Classes

We need some operators to explore the relations between structured objects – tree Token classes – in TS. All results of following operation are unique up to isomorphism of objects.

For any p-classes $T = \langle B, E, \Upsilon \rangle$, $T_1 = \langle B_1, E, \Upsilon_1 \rangle$, $T_2 = \langle B_2, E, \Upsilon_2 \rangle$ and any p-map $\langle f, 1_E, p \rangle : T_1 \to T_2$, we define following operations as

Basic Set Theory t-Classes Operations

The basic set theory t-classes operations are:

- *merging* $\Cup$ defined as

$$T \Cup T^{'} = \left\langle \mathbf{base}(T) \cup \mathbf{base}\left(T^{'}\right), E, \mathbf{heap}(T) \cup \mathbf{heap}\left(T^{'}\right) \right\rangle;$$

- *meeting* $\Cap$ defined as

$$T \Cap T^{'} = \left\langle \mathbf{base}(T) \cap \mathbf{base}\left(T^{'}\right), E, \mathbf{heap}(T) \cap \mathbf{heap}\left(T^{'}\right) \right\rangle;$$

- *forgetting* $\varnothing$ defined as

$$\varnothing T = \langle \mathbf{base}(T), E, \varnothing \rangle,$$

  and

$$\varnothing \left(\langle f, 1_E, p \rangle\right) = \langle f, 1_E, \varnothing \rangle;$$

- *stuffing* $\circledast$ defined as

$$\circledast \left(\langle B, E, \Upsilon \rangle\right) = \left\langle B, E, \mathcal{T}\left(B \coprod \{E\}\right) \right\rangle;$$

- *deleting* $\ominus$ defined as

$$T_2 \ominus T_1 = \langle \mathbf{base}\left(T_2\right), E, \mathbf{heap}\left(T_2\right) - \mathbf{heap}\left(T_1\right) \rangle.$$

Functor operations and related p-maps:

- **Introducing unknown** $\odot$: Let $\varepsilon, e \notin B \cup E$,

$$\odot \left(\langle B, E, \Upsilon \rangle\right) = \langle B \cup \{e\}, E \cup \{\varepsilon\}, \Upsilon \cup \{(\varepsilon, e)\} \rangle$$

  and

$$\odot \left(\langle f, 1_E, p \rangle\right) = \langle f \cup \{e \mapsto e\}, E \cup \{\varepsilon \mapsto \varepsilon\}, \Upsilon \cup \{(\varepsilon, e)\} \rangle.$$

  We say that e is a *unknown element* introduced in eT.

- **Powering** ∇: Let $\gamma, \varepsilon, e \notin B \cup E$, $\mathcal{P}(B)$ be the power set of B, then

$$\nabla(\langle B, E, \Upsilon\rangle) = \langle \{e\} \cup B \cup \mathcal{P}(B), E \cup \{\varepsilon, \gamma\}, \Upsilon_T\rangle$$

where

$$\Upsilon_T = \{(\gamma, S, x) \,| S \in \mathcal{P}(B), x \in S or x = e \ \} \cup$$

$$\left\{(x_1, x_2, \cdots, x_n) \middle| \left(x_1^{'}, x_2^{'}, \cdots, x_n^{'}\right) \in \Upsilon, x_i^{'} \in x_i or x_i^{'} = x_i \ \right\}$$

and

$$\nabla(\langle f, 1_E, p\rangle) = \left\langle f \cup \{e \mapsto e\} \cup \mathcal{P}(f), 1_E \cup \{\varepsilon \mapsto \varepsilon, \gamma \mapsto \gamma\}, p^{'} \right\rangle$$

where

$$p^{'} = p \cup \mathbf{on}\left(\mathcal{M}\left((f \cup \{e \mapsto e\} \cup \mathcal{P}(f))^{*}\right), \Upsilon_{\nabla T_1}, \Upsilon_{\nabla T_2}\right).$$

- **Obscuring** $\rightsquigarrow$: obscuring $T = \langle B, E, \Upsilon\rangle$ is a p-class

$$\rightsquigarrow(T) = \left\langle B \coprod E, \varnothing, \Upsilon \right\rangle$$

and

$$\rightsquigarrow(\langle f, 1_E, p\rangle) = \left\langle f \coprod 1_E, \varnothing, p \right\rangle$$

Binary operations and related p-maps:

- **Matchup** $\otimes$: (*product*)

$$\langle B_1, E, \Upsilon_1\rangle \otimes \langle B_2, E, \Upsilon_2\rangle = \langle B_1 \times B_2, E, \Upsilon\rangle$$

where for any

$$(\langle x_1, y_1\rangle, \langle x_2, y_2\rangle, \cdots, \langle x_n, y_n\rangle)$$

with $\langle x_i, y_i\rangle$ nominally denote x_i for $x_i = y_i \in E$,

$$(\langle x_1, y_1\rangle, \langle x_2, y_2\rangle, \cdots, \langle x_n, y_n\rangle) \in \Upsilon$$

iff $(x_1, x_2, \cdots, x_n) \in \Upsilon_1$ and $(y_1, y_2, \cdots, y_n) \in \Upsilon_2$.

Example 2. *Given an ordered set* $\mathbf{P} = \langle P, \geqslant\rangle$*, we can use a class* $T_{\mathbf{P}} = \langle P, \{\geqslant\}, \Upsilon\rangle$ *to represent* $\mathbf{P}$*, where* $B = P$*, and*

$$\Upsilon = \{(\geqslant, x, y) \,| x \geqslant ; x, y \in P \ \}.$$

Given two ordered sets $\mathbf{P} = \langle P, \geqslant \rangle$ *and* $\mathbf{P}' = \left\langle P', \geqslant \right\rangle$*, the product of* $T_{\mathbf{P}} \otimes T_{\mathbf{P}'}$ *is naturally the correspondent class of product* $\mathbf{P} \times \mathbf{P}' = \left\langle P \times P', \geqslant \right\rangle$*, where* $\left(P \times P'\right)$ *is the Cartesian product of* P *and* P'*, and*

$$(\langle x_1, x_2 \rangle, \langle y_1, y_2 \rangle) \in r_{T_{\mathbf{P}} \otimes T_{\mathbf{P}'}}$$

iff $x_1 \geqslant y_1$ *and* $x_2 \geqslant y_2$*.*

- **Blending** $\oplus$: (*extending product*)

$$\langle B_1, E, \Upsilon_1 \rangle \oplus \langle B_2, E, \Upsilon_2 \rangle = \langle B_1 \times B_2, E, \Upsilon \rangle$$

  where for any

$$(\langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle, \cdots, \langle x_n, y_n \rangle)$$

  with $\langle x_i, y_i \rangle$ nominally denote x_i for $x_i = y_i \in E$,

$$(\langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle, \cdots, \langle x_n, y_n \rangle) \in \Upsilon$$

  iff $(x_1, x_2, \cdots, x_n) \in \Upsilon_1$ or $(y_1, y_2, \cdots, y_n) \in \Upsilon_2$.

- **Union** $\uplus$: (*coproduct* or *Y-union*) given two p-classes $T_1 = \langle B_1, E, \Upsilon_1 \rangle$ and $T_2 = \langle B_2, E, \Upsilon_2 \rangle$, and for some $\varepsilon \notin B_1 \cup B_2 \cup E$, let

$$B = B_1 \times \{\varepsilon\} \cup \{\varepsilon\} \times B_2$$

  be just isomorphic to copruduct $B_1 \coprod B_2$ in **Set**, and the inclusion maps in $\mathcal{U}(\mathbf{Set} \times \mathbf{E})$ are as in diagram

$$\mathcal{U}(\langle B_1, E \rangle) \overset{inc_1}{\rightarrow} \mathcal{U}\left(\left\langle B_1 \coprod B_2, E \right\rangle\right) \overset{inc_2}{\leftarrow} \mathcal{U}(\langle B_2, E \rangle).$$

  The union

$$\langle B_1, E, \Upsilon_1 \rangle \uplus \langle B_2, E, \Upsilon_2 \rangle = \left\langle B_1 \coprod B_2, E, \Upsilon \right\rangle$$

  where $(x_1, x_2, \cdots, x_n) \in \Upsilon$ iff (1) $x_i = inc_1(y_i)$ and $(y_1, y_2, \cdots, y_n) \in \Upsilon_1$, or (2) $x_i = inc_2(y_i)$ and $(y_1, y_2, \cdots, y_n) \in \Upsilon_2$.

- **Deleting** $\ominus$: if $T_1 = \langle B_1, E, \Upsilon_1 \rangle$ is a subclass of $T_2 = \langle B_2, E, \Upsilon_2 \rangle$

$$T_2 \ominus T_1 = \langle B_2, E, \Upsilon_2 - \Upsilon_1 \rangle.$$

- **Refering** $\lhd$: (*exponent object*) given two p-classes $T_1 = \langle B_1, E, \Upsilon_1 \rangle$ and $T_2 = \langle B_2, E, \Upsilon_2 \rangle$,

$$T_1 \lhd T_2 = \left\langle (B_1)^{B_2}, E, \Upsilon \right\rangle$$

  for any $\zeta_1, \zeta_2, \cdots, \zeta_m \in (B_1)^{B_2} \cup E$, if for any

$$\langle \iota_1, \iota_2, \cdots, \iota_m \rangle \in \Upsilon_2$$

  such that for any j, $\zeta_j \in E$ or $\iota_j \in E$ implies $\iota_j = \zeta_j$, there is

$$\langle \zeta_1(\iota_1), \zeta_2(\iota_2), \cdots, \zeta_m(\iota_m) \rangle \in \Upsilon_1$$

(1)

where $\zeta_j(\iota_j)$ nominally denotes ζ_j for $\iota_j = \zeta_j \in E$, then $\langle \zeta_1, \zeta_2, \cdots, \zeta_m \rangle \in \Upsilon$; otherwise, $\langle \zeta_1, \zeta_2, \cdots, \zeta_m \rangle \notin \Upsilon$.

Operations with functions and p-maps:

- **Lifting** $\Uparrow$: given a set $S \subset B$,

$$\Uparrow (\langle B, E, \Upsilon \rangle, S) = \left\langle B - S, E \coprod S, \left\{ \left(inc_{S \to E \coprod S}(x) \mapsto x \right) | x \in S \ \right\} \Upsilon \right\rangle.$$

- **Absolute lifting** $\Uparrow\!\Uparrow$: given a set $S \subset B$,

$$\Uparrow\!\Uparrow (\langle B, E, \Upsilon \rangle, S) = \left\langle B, E \coprod S, \Upsilon \cup \left\{ \left(inc_{S \to E \coprod S}(x), x \right) | x \in S \ \right\} \right\rangle.$$

- **Release** $\Downarrow$: given a set $E^{'} \subset E$,

$$\Downarrow \left(\langle B, E, \Upsilon \rangle, E^{'} \right) = \left\langle B \coprod E^{'}, E - E^{'}, \Upsilon \right\rangle.$$

- **Renaming structure** $\oslash$: given an injective function $\alpha : E_1 \to E_2$, $\oslash (\langle B, E_1, \Upsilon \rangle, \alpha) = \langle B, E_2, (1_B \coprod \alpha)(\Upsilon) \rangle$.

- **Reversing** $\rightleftharpoons$: given two p-classes $T_1 = \langle B_1, E, \Upsilon_1 \rangle$, $T_2 = \langle B_2, E, \Upsilon_2 \rangle$ and p-map $f : T_1 \to T_2$, $\rightleftharpoons (f)$ is a p-map from $\nabla \varnothing T_2$ to $\nabla \varnothing T_1$ such that

$$\rightleftharpoons (f)(y) = \begin{cases} f^{-1}(y), y \in \mathcal{P}(B_2) \\ \quad e, y \notin \mathcal{P}(B_2) \end{cases}.$$

- **Generalizing** $\rtimes$: let $T^{'} = \left\langle B^{'}, E, \Upsilon^{'} \right\rangle$ be a subclass of $T = \langle B, E, \Upsilon \rangle$, and $f : T \to \nabla T$ be a p-map, if $x \in B$ implies $x \in f(x)$ or $x = f(x)$, then

$$\rtimes \left(f, T^{'} \right) = f \left(T^{'} \right)$$

  else

$$\rtimes \left(f, T^{'} \right) = f \left(\varnothing T^{'} \right).$$

Example 3. *We consider the n-dimensional real-valued vector space $V = X_1 \times X_2 \times ... \times X_n$. For each X_i, $i = 1, 2, \cdots, n$, we consider the ordered set $\mathbf{P}_i^0 = \langle X_i, \geq \rangle$ and its dual $\mathbf{P}_i^1 = \langle X_i, \leq \rangle$. There are 2^n order relations on $X_1 \times X_2 \times ... \times X_n$:*

$$\mathbf{M}_j = \mathbf{P}_1^{j_1} \times \mathbf{P}_2^{j_2} \times \cdots \times \mathbf{P}_n^{j_n} = \langle X_1 \times X_2 \times ... \times X_n, \geqslant_j \rangle,$$

where j_i is either 0 or 1, and $j = 0, 1, 2, \cdots, 2^{n-1}$. Therefore, there are 2^n Token classes products

$$T_{\mathbf{M}_j} = \left(\oslash \left(T_{\mathbf{P}_1^{j_1}}, f_1 \right) \right) \otimes \left(\oslash \left(T_{\mathbf{P}_2^{j_2}}, f_2 \right) \right) \otimes \cdots \otimes \left(\oslash \left(T_{\mathbf{P}_n^{j_n}}, f_n \right) \right)$$

where $f_i : \{\geq^{j_i}\} \to \{\geqslant_j\}$ is the function $f(\geq^{j_1}) = \geqslant_j$ with $\geq^0$ is $\geq$ and $\geq^1$ is $\leq$.

Example 4. *We consider the structures on data set. Let $\mathcal{D}$ is a collection data instances of which each refere to a n-dimension real-valued vector in $X_1 \times X_2 \times ... \times X_n$. Each instance in $\mathcal{D}$ is a tuple (k, v), where $v = (x_1, x_2, \cdots, x_n)$ and $k \in \{1, 2, \cdots, l\}$ is the id of the instance, and l is the size of $\mathcal{D}$. Let*

$T_I = \langle \mathbb{N}, \varnothing, \varnothing \rangle$

then $T_I \oplus T_{\mathbf{M}_j}$ represents a weak ordered set. *Now each instance in $\mathcal{D}$ is also an object in $T_I \oplus T_{\mathbf{M}_j}$. Then $D_j = \langle \mathcal{D}, \{\geqslant_j\}, \Upsilon \rangle$ is a subclass of $T_I \oplus T_{\mathbf{M}_j}$, where*

$\Upsilon = r_{T_I \oplus T_{\mathbf{M}_j}} \cap \mathcal{D}^2$

Giving j-th covariant Graph $\mathbf{G}_j = \langle \mathcal{D}, E_j \rangle$, which represents the diagram of weak order D_j, let $H_j = \left\langle \mathcal{D}, \{\geqslant_j\}, \Upsilon_{H_j} \right\rangle$ is a class such that

$$(\geqslant_j, (k_1, v_1), (k_2, v_2)) \in \Upsilon_{H_j}$$

iff $(v_1, v_2) \in E_j$. Then H_j is a subclass of D_j . We also use $\mathbf{GC}_j$ to denote $\% (H_j, f)$ where $f : \{\geqslant_j\} \to \{\succcurlyeq_j\}$ is the function $f(\geqslant_j) = \succcurlyeq_j$. We call $\mathbf{GC}_j$ a covariant p-class *of $\mathcal{D}$.*

Example 5. *Let $\langle \mathcal{D}, f, \mathcal{C} \rangle$ be the training data, $\mathcal{C}$ is the set of labels, and $f \colon \mathcal{D} \to \mathcal{C}$ is a function which assigns each vector in $\mathcal{D}$ a label in $\mathcal{C}$. Then*

$$T_{\mathcal{C}} = \langle \mathcal{C}, \{\succcurlyeq\}, \Upsilon_{\mathcal{C}} \rangle$$

is a Token class of discrete (weak) ordered set, where

$$r_{\mathcal{C}} = \{(\succcurlyeq, C, C) | C \in \mathcal{C}\}$$

Given a test instance $d = \langle l + 1, v \rangle$, let $\mathbf{GC}'_j$, $j = 0, 1, 2, \cdots, 2^n - 1$, are the all 2^n covariant p-classes on $\mathcal{D} \cup \{d\}$, we can extends f to

$$p_i \colon \mathcal{D} \cup \{d\} \to T_{\mathcal{C}},$$

$i = 1, 2, \cdots, |\mathcal{C}|$, and each p_i gives a label $p_i(d)$ for d. Let

$$\mathbf{GC} = \oslash \left(\mathbf{GC}'_0, g_1 \right) \uplus \oslash \left(\mathbf{GC}'_1, g_2 \right) \uplus \cdots \uplus \oslash \left(\mathbf{GC}'_{2^n - 1}, g_{2^n - 1} \right)$$

where $g_j : \{\succcurlyeq_j\} \to \{\succcurlyeq\}$ is the function $f(\succcurlyeq_j) = \succcurlyeq$. Note the union $\uplus$ is defined as:

$$\langle B_1, E, \Upsilon_1 \rangle \uplus \langle B_2, E, \Upsilon_2 \rangle = \left\langle B_1 \coprod B_2, E, \Upsilon \right\rangle$$

where $(x_1, x_2, \cdots, x_n) \in \Upsilon$ iff $(x_1, x_2, \cdots, x_n) \in \Upsilon_1$ or $(x_1, x_2, \cdots, x_n) \in \Upsilon_2$, in addition that $\langle x_i, y_i \rangle$ nominally denote x_i if $x_i = y_i \in E$.

9 Reification of Tree Token Classes

Reification is an operation which can represent tree Token classes as p-classes and build one to one correspondence between t-maps and p-maps.

Definition 15. *The* tree *functor $\Xi : TS \to TTS$ is defined as*

$$\Xi(\langle B, E, \Upsilon \rangle) = \langle B, E, \Upsilon \rangle .$$

Suppose $\Finv, \varepsilon \in$ **terms**.

Definition 16. *For a tree Token $r \in \langle B, E, \Upsilon \rangle$,*

$$\partial(r) = \{.\, \{\varepsilon \cdot r\}, \quad r \in B \coprod E \left\{\varepsilon \cdot r, \&^{-1}(r)\right\}, \mathbf{tary}(r) \neq 0$$

is called a self nesting *of r; a* reification *of tree Token r is a p-class*

$$\Finv(r) = \langle \ll^* (r), E, \Upsilon \rangle$$

where

$$\Upsilon = \left\{\varepsilon \cdot p \cdot \&^{-1}(p) \,|p \in (\ll^* (r) - B) \ \right\}$$

$$\cup \partial(r).$$

From this definition, it follows:

Lemma 4. *Taking any set E as core, if there is a t-map $g : \{r_1\} \to \{r_2\}$, then for any $x \in \ll^* (r_1)$, we have* $\mathbf{tmorph}(g)(x) \in \ll^* (r_2)$.

Clear.

Lemma 5. *Taking any set E as core, there is at most one p-map between $\Finv(r_1)$ and $\Finv(r_2)$.*

Clear.

Proposition 15. *Taking any set E as core, there is a t-map between $\{r_1\}$ and $\{r_2\}$ iff there is a p-map between $\Finv(r_1)$ and $\Finv(r_2)$.*

Proof. Suppose there is a t-map $g : \{r_1\} \to \{r_2\}$. Let

$$f = \mathbf{tmorph}(g).$$

Note $f(r_1) = g_0(r_1) = r_2$, hence it folows $f(r_1) \in \Finv(r_2)$. For If

$$r_1 = (x_1, x_2, \cdots, x_n)$$

and

$$\mathbf{base}\,(r_1) = \{x_1, x_2, \cdots, x_n\}$$

then

$$\Finv(r_1) = \langle \{x_1, x_2, \cdots, x_n\}, E, \{r_1\} \rangle$$

and

$$f(r_1) = f((x_1, x_2, \cdots, x_n))$$

$$= \left(f\left(x_1\right), f\left(x_2\right), \cdots, f\left(x_n\right)\right)$$

$$= \left(g_0\left(x_1\right), g_0\left(x_2\right), \cdots, g_0\left(x_n\right)\right)$$

$$= g_0\left(r_1\right).$$

It follows that there is just a t-map from $\dashv\left(r_1\right)$ to $\dashv\left(r_2\right)$ extending

$$\left\{x \mapsto f(x)\left|x \in \left\{x_1, x_2, \cdots, x_n\right\}\right.\right\}.$$

If

$$\left(r^{'}, x_1, x_2, \cdots, x_n\right) \in \dashv\left(r_1\right)$$

and $x_1, x_2, \cdots, x_n \in \mathbf{base}\left(r_1\right)$, then

$$f\left(r^{'}\right) = f\left(\left(x_1, x_2, \cdots, x_n\right)\right)$$

$$= \left(f\left(x_1\right), f\left(x_2\right), \cdots, f\left(x_n\right)\right)$$

$$= \left(g_0\left(x_1\right), g_0\left(x_2\right), \cdots, g_0\left(x_n\right)\right)$$

$$= \left(g\left(x_1\right), g\left(x_2\right), \cdots, g\left(x_n\right)\right)$$

$$= g\left(r^{'}\right).$$

Note $g\left(r^{'}\right) \in \ll^{*}\left(r_2\right)$. Hence $g\left(r^{'}\right) \in \mathbf{base}\left(\dashv\left(T_2\right)\right)$, also $f\left(r^{'}\right) \in \mathbf{base}\left(\dashv\left(T_2\right)\right)$. Hence

$$\left(\varepsilon, f\left(r^{'}\right), f\left(x_1\right), f\left(x_2\right), \cdots, f\left(x_n\right)\right) \in \dashv\left(r_1\right)$$

It follows, inductively on depth of r, that

$$f(r) \in \mathbf{base}\left(\dashv\left(T_2\right)\right)$$

and $\partial\left(f(r)\right) \in \dashv\left(T_2\right)$ hold for any $r \in \left(\dashv\left(r_1\right) - \left\{r_1\right\}\right)$. Therefore, there is a p-map from $\dashv\left(r_1\right)$ to $\dashv\left(r_2\right)$ extending

$$\left\{x \mapsto f(x)\left|x \in \mathbf{base}\left(\dashv\left(T_1\right)\right)\right.\right\}.$$

Conversely, if there is a p-map

$$f : \dashv\left(r_1\right) \to$$

$\dashv\left(r_2\right)$, because only r_1 and r_2 do not contain ε, it easily seen that

$$f\left(r_1\right) = r_2.$$

Therefore, there is a t-map $g : r_1 \to r_2$ extends

$$\{x \mapsto f(x)\,|x \in \mathbf{base}\,(r_1)\ \}\,.$$

The Proposition is proved. $\square$

Definition 17 (Reification)**.** *A* reification *of* $T = \langle B, E, \Upsilon\rangle$ *is*

$$\dashv(T) = \left\langle_{r\in\Upsilon}\mathbf{base}\,(\dashv(r))\,, E \cup \{\dashv\}\,, _{r\in\Upsilon}\mathbf{heap}\,(\dashv(r))\right\rangle$$

and for t-map $\langle f_0, 1_E, p\rangle : T_1 \to T_2$*,*

$$\dashv(\langle f_0, 1_E, p\rangle) = \left\langle f_0', 1_E, p\right\rangle.$$

where

$$f_0' = \left\{x \mapsto \mathbf{tmorph}\left(f_0 \coprod 1_E\right)(x)\,|x \in \mathbf{base}\,(\dashv(T_1))\ \right\}$$

and

$$p = \left\{r \mapsto \mathbf{tmorph}\left(f_0 \coprod 1_E\right)(r)\,|r \in \mathbf{heap}\,(\dashv(T_1))\ \right\}.$$

Note that if $\dashv(\langle B, E, \Upsilon\rangle)$ *is not defined. For the simplicity, we use* $\dashv(\langle B, E, \Upsilon\rangle)$ *and* $\dashv(f)$ *to denote* $\dashv(\langle B, E, \Upsilon\rangle\,, \dashv)$ *and* $\dashv(f, \dashv)$ *if* $\dashv \notin B \coprod E$*.*

Proposition 16. *For any* $r = (x_1, x_2, \cdots, x_2) \in T$ *and any Tokens map* $f : T \to T'$*, we have*

$$f(r) = \dashv(f)(r).$$

Proof. Because $r \in \dashv(T)$, then $\dashv(f)(r) \in \mathbf{heap}\left(\dashv\left(T'\right)\right)$. This is simple. $\square$

Proposition 17. *Given any two tree Token classes* T_1 *and* T_2*, then*

$$TPC\,(T_1, T_2) \cong TPC\,(\dashv(T_1)\,, \dashv(T_2)) \cong TS\,(\dashv(T_1)\,, \dashv(T_2))\,.$$

Proof. Clear. $\square$

10 Conclusion

Throughout this paper, we have elucidated that the Token Space constitutes a bi-Cartesian closed category, possessing all finite limits and encompassing categorical structures adept at classifying subobjects. This foundation not only bolsters the interpretability and theoretical comprehension of deep learning models but also furnishes a robust framework for exploring novel computational constructs within AI research.

Despite these advancements, there remain several avenues of investigation yet to be explored. For example:

- Any category of structured objects hardly can use a single pullback square as subclass classifier.
- Let $\mathcal{C}_{Cartisain}$ and $\mathcal{C}_{Topos}$ are two *supper operators* on subcategories of Token categories such that for any subcategory $\mathbf{P}$ of a Token category $\mathbf{PC},$ $\mathcal{C}_{Cartisain}(\mathbf{P})$ and $\mathcal{C}_{Topos}(\mathbf{P})$ are minimum bicartisain closed subcategory and topos subcategories, of which $\mathbf{P}$ is a subcategory, of $\mathbf{PC}.$ Naturally, a problem arise. Given a concrete category $\mathbf{C}$, How does $\mathcal{R}ep_{\mathbf{C}}(\mathbf{C})$ relate to
$$\mathcal{C}_{Cartisain}(\mathcal{R}ep_{\mathbf{C}}(\mathbf{C}))$$
and $\mathcal{C}_{Topos}(\mathcal{R}ep_{\mathbf{C}}(\mathbf{C}))$?

This inquiry into the structural and operational dynamics of Token Space not only paves the way for deeper theoretical insights but also opens the door to more sophisticated and interpretable machine learning models. The journey of integrating category theory with computational models is far from complete, and the questions we have outlined offer fertile ground for further exploration. As we continue to unravel the complexities of Token Space, we anticipate uncovering novel strategies for model construction, analysis, and optimization, further advancing the frontier of AI research.

References

[1] F. William Lawvere and Stephen H. Schanuel, *Conceptual Mathematics: A First Introduction to Categories*, Cambridge University Press, 1997.

[2] Steve Awodey, *Category Theory*, Oxford University Press, 2006.