Electronic Notes in Theoretical Computer Science 53 (2002)
URL: http://www.elsevier.nl/locate/entcs/volume53.html 12 pages

Analogy and Formal Languages

Yves Lepage [1,2]

ATR Spoken Language Translation Research Laboratories
619-0288 Kyōto, Japan

Abstract

In this paper, we advocate a study of analogies between strings of symbols for their own sake. We show how some sets of strings, *i.e.,* some formal languages, may be characterized by use of analogies. We argue that some preliminary "good properties" obtained may plead in favour of the use of analogy in the study of formal languages in relationship with natural language.

1 Introduction

In linguistics, for Humboldt, Hermann Paul, Baudoin de Courtenay and Saussure, analogy applied to words is the cognitive process by which, given two forms of a word, and only one form of a second word, one creates the missing form: *ōrātōrem* : *ōrātor* = *honōrem* : x ⇒ x = *honor* (analogical equation). Such analogies between strings of symbols, generally noted $A : B = C : D$, put four strings of symbols into "proportions." They seem to be independent of any particular language, and the faculty of solving such equations, at least on the symbol level, has been argued to be an autonomous process and a universal ability [5].

We shall be concerned only with analogies on the symbol level, and not directly with analogies like *arm* : *hand* = *leg* : *foot*. Moreover, we will not be concerned with sentences like *an atom is like the solar system*, which are improperly called analogies precisely because they rest on analogies (here: *an electron is to the nucleus as a planet is to the sun*, see [3]).

[1] Thanks to Prof. Boitet for reading drafts of this paper. Also, thanks to the anonymous referees who pointed out many imperfections and mistakes. All remaining errors are, of course, mine.
[2] Email: yves.lepage@atr.co.jp

2 Analogies

2.1 Examples

So we consider analogies only pertaining to the level of symbols. Let us clarify with some cases that do not exemplify the type of analogies that we intend to deal with:

$$\square : \bigcirc \neq \blacksquare : \bullet$$

$$abc : ac \neq acb : ac$$

$$a : b \neq c : d$$

$$walk : walked \neq go : went$$

$$bird : wings \neq fish : fins$$

$$abc : ac \neq ca : anything$$

The first line shows that symbols cannot be decomposed: in our view, $\bullet$ is not $\bigcirc$ plus black. The second line shows that, for us, analogy deals firstly with symbols, and only in a second step with order, so that in this case, the valid analogy for us is: $abc : ac = acb : ab$. The following three lines show that no further information outside the given will be considered: a resolution algorithm should not have knowledge about, say, the alphabetical order, nor should it know English conjugation, nor should it have knowledge about animal morphology. The last line, and many other similar ones, show that, if a single symbol of the first string (here b) does not belong to either the second string or to the third, then, there is no analogy which can hold, so that the following analogical equation does not have any solution: $abc : ac = ca : \mathtt{x}$.

Now, the following examples are, in our view, valid analogies. Although the third one yields a barbarism (*goed*), it is the exact solution to the analogical equation $walk : walked = go : \mathtt{x}$ (see [15, p. 231]). The last one shows that analogical equations may have several solutions. Hence, in general, a given analogical equation may have no solution, a unique solution, or several solutions.

$$\square\bigcirc : \triangle\bigcirc = \square\bullet : \mathtt{x} \quad \Rightarrow \quad \mathtt{x} = \triangle\bullet$$

$$look : looked = walk : \mathtt{x} \quad \Rightarrow \quad \mathtt{x} = walked$$

$$walk : walked = go : \mathtt{x} \quad \Rightarrow \quad \mathtt{x} = goed$$

$$fable : fabulous = miracle : \mathtt{x} \quad \Rightarrow \quad \mathtt{x} = miraculous$$

$$ab : aabb = aaaabbbb : \mathtt{x} \quad \Rightarrow \quad \mathtt{x} = aaaaabbbbb$$

$$aba : aa = cbcbcbc : \mathtt{x} \quad \Rightarrow \quad \mathtt{x} = ccbcbc \vee \mathtt{x} = cbccbc \vee \mathtt{x} = cbcbcc$$

2.2 Formalization

Starting with general properties, Aristotle (*Nicomachian Ethics*), as well as Hermann Paul [13, chap. V, § 76, p. 107] use common sense when they perform what is called exchange of the means: $A : B = C : D \Leftrightarrow A : C = B : D$. Another property, relying on the symmetry of equality, is also used by Aristotle: $A : B = C : D \Leftrightarrow C : D = A : B$. These two properties considered as axioms, lead to five other equivalent forms of the same analogy, so that one can give the following theorem, useful in obtaining further theorems. The proof is trivial.

Theorem 2.1 *The eight following analogies are equivalent:*

$$
\begin{aligned}
&A : B = C : D \\
&A : C = B : D \quad \textit{(exchange of the means)} \\
&B : A = D : C \quad \textit{(inversion of ratios)} \\
&B : D = A : C \\
&C : A = D : B \\
&C : D = A : B \quad \textit{(symmetry of equality)} \\
&D : B = C : A \quad \textit{(exchange of the extremes)} \\
&D : C = B : A \quad \textit{(symmetry of reading)}
\end{aligned}
$$

2.2.1 Similarity and Distances

Having examined a range of analogies between strings of symbols, [3] we conjectured that no solution to the analogical equation $A : B = C : \mathtt{x}$ exists if one of the symbols in A appears in neither B nor C (see 2.1). By contrapositive, for a solution to exist, any symbol in A should appear in B or C or in both, so that the following axiom can be laid.

AXIOM **1** *Let $\mathcal{V}$ be an alphabet.*

$$\forall (A,B,C,D) \in (\mathcal{V}^*)^4, \quad A : B = C : D \quad \Rightarrow \quad \overline{A} \subset \overline{B} \cup \overline{C}$$

This should hold also when taking the order of symbols into account, so that, in general, analogy does not reorder any of the components of the strings. As a consequence, the length of A is less than or equal to the sum of its similarities with B and C, where the similarity $\sigma(A,B)$ between two strings of symbols A and B is defined as the length of their longest common subsequence. [4]

[3] We call $\mathcal{V}$ the set of symbols, and $\mathcal{V}^*$ the set of strings built with elements of $\mathcal{V}$. In the sequel, thus, $(A,B,C,D) \in (\mathcal{V}^*)^4$. $\overline{A}$ denotes the set of different symbols in the string A.

[4] Recall that a subsequence is not necessarily connected. For instance, $\sigma(abababa, aaaaa) = 5$, or $\sigma(triangle, angularity) = 4$.

AXIOM **2** *Let $\mathcal{V}$ be an alphabet.*

$$\forall (A,B,C,D) \in (\mathcal{V}^*)^4, \quad A:B=C:D \quad \Rightarrow \mid A \mid \leq \sigma(A,B)+\sigma(A,C)$$

If the length of A is less than this sum, it means that some symbols in A appear in the same order also in B and C, and hence also in D, as they must be copied in the same order in the solutions of the analogical equation. Let us call $\gamma(A,B,C,D)$ the number of occurrences of such symbols. One gets:

$$A:B=C:D \quad \Rightarrow \quad \mid A \mid = \sigma(A,B)+\sigma(A,C)-\gamma(A,B,C,D)$$

By application of Theorem 2.1, one gets seven such equalities:

$$\begin{aligned}
\mid A \mid &= \sigma(A,C)+\sigma(A,B)-\gamma(A,C,B,D)\\
\mid B \mid &= \sigma(B,A)+\sigma(B,D)-\gamma(B,A,D,C)\\
\mid B \mid &= \sigma(B,D)+\sigma(B,A)-\gamma(B,D,A,C)\\
\mid C \mid &= \sigma(C,A)+\sigma(C,D)-\gamma(C,A,B,D)\\
\mid C \mid &= \sigma(C,D)+\sigma(C,A)-\gamma(C,D,A,B)\\
\mid D \mid &= \sigma(D,B)+\sigma(D,C)-\gamma(D,B,C,A)\\
\mid D \mid &= \sigma(D,C)+\sigma(D,B)-\gamma(D,C,B,A)
\end{aligned}$$

All γ's being equal and by the symmetry of similarity, the list of equalities reduces to only four ones with $\mid A \mid$, $\mid B \mid$, $\mid C \mid$ and $\mid D \mid$ as their first members.

$$\begin{cases}
\mid A \mid = \sigma(A,B)+\sigma(A,C)-\gamma(A,B,C,D)\\
\mid B \mid = \sigma(B,A)+\sigma(B,D)-\gamma(A,B,C,D)\\
\mid C \mid = \sigma(C,A)+\sigma(C,D)-\gamma(A,B,C,D)\\
\mid D \mid = \sigma(D,B)+\sigma(D,C)-\gamma(A,B,C,D)
\end{cases}$$

This leads easily to the following lemma:

Lemma 2.2 *Let $\mathcal{V}$ be an alphabet.* $\forall (A,B,C,D) \in (\mathcal{V}^*)^4$,

$$A:B=C:D \quad \Rightarrow \quad \begin{cases}
\mid A \mid - \sigma(A,B) = \mid C \mid - \sigma(C,D) & (1)\\
\mid B \mid - \sigma(B,D) = \mid A \mid - \sigma(A,C) & (2)\\
\mid C \mid - \sigma(C,A) = \mid D \mid - \sigma(D,B) & (3)\\
\mid D \mid - \sigma(D,C) = \mid B \mid - \sigma(B,A) & (4)
\end{cases}$$

By addition of lines (1) and (4), and by the symmetry of similarity, one gets the following remarkable result, which says that the sums of the lengths of the extremes and of the means are equal.

Lemma 2.3 *Let $\mathcal{V}$ be an alphabet.*

$$\forall (A,B,C,D) \in (\mathcal{V}^*)^4, \quad A:B=C:D \quad \Rightarrow \mid A \mid + \mid D \mid = \mid B \mid + \mid C \mid$$

As a consequence of the previous result, there is no place for general reduplication in this framework. General reduplication would have led us to write down: *Let $\mathcal{V}$ be an alphabet.*

$$\forall (A,B) \in (\mathcal{V}^*)^2, \quad A:B=AA:BB$$

But this contradicts the previous lemma. For example, $a^n : b^m = a^{2n} : b^{2m}$, would imply $n+2m=2n+m$, which is possible if and only if $n=m$. [5]

The previous results can be gathered in the following (partial) characterization of analogies between strings of symbols, where D appears only on the left side of the equalities, and where the right sides contain only A, B and C.

Theorem 2.4 *Let $\mathcal{V}$ be an alphabet.* $\forall (A,B,C,D) \in (\mathcal{V}^*)^4$,

$$A:B=C:D \quad \Rightarrow \quad \begin{cases} \sigma(B,D) = -\mid A \mid + \mid B \mid + \sigma(A,C) \\ \sigma(C,D) = -\mid A \mid + \mid C \mid + \sigma(A,B) \\ \mid D \mid = -\mid A \mid + \mid B \mid + \mid C \mid \\ \gamma(A,B,C,D) = -\mid A \mid + \sigma(A,B) + \sigma(A,C) \end{cases}$$

With one more constraint, this characterization leads directly to the implementation of an algorithm for the resolution and the verification of analogical equations. However, this partial characterization is sufficient for the sequel of this paper, in order for us to prove some other formal results.

It is known that the similarity is in relationship with a particular string distance [10], which we call *edit distance*, to follow the terminology of [18, p. 40–41], the one equipped only with insertions and deletions. [6]

Proposition 2.5 *Let $\mathcal{V}$ be an alphabet of symbols, let σ be the similarity between strings and let δ be the edit distance.*

$$\forall (A,B) \in (\mathcal{V}^*)^2, \quad \delta(A,B) = \mid A \mid + \mid B \mid - 2 \times \sigma(A,B)$$

By substituting the similarities with the distances in Theorem 2.4, one gets a more intuitive form of the previous characterization of analogies

[5] In fact, it appears that "pure analogy" and "reduplication analogy" are to be considered as two different axiomatic systems, the first one with exchange of the means, symmetry of equality and our two previous axioms, and the second one with exchange of the means, symmetry of equality and a reduplication axiom.

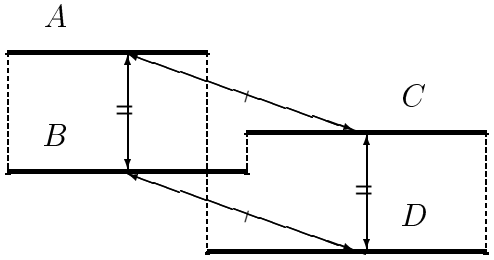
[6] Substitutions are obtained by applying a deletion and then an insertion, or an insertion followed by a deletion. For instance, $\delta(edit, edition) = 3$ (insertion of three symbols: *i, o* and *n*), $\delta(triangle, angularity) = 10$ (deletion of *t, r, i,* insertion of *u,* deletion of *e,* and insertion of *a, r, i, t, y*). The edit distance is a metric, as it verifies the three axioms of identity, symmetry and triangular inequality.

between strings of symbols.

Theorem 2.6 *Let $\mathcal{V}$ be an alphabet.* $\forall(A,B,C,D) \in (\mathcal{V}^*)^4$,

$$A:B=C:D \quad \Rightarrow \quad \begin{cases} \delta(A,B) = \delta(C,D) \\ \delta(A,C) = \delta(B,D) \\ \mid A \mid + \mid D \mid = \mid B \mid + \mid C \mid \\ \gamma(A,B,C,D) = \frac{1}{4} \times (\mid A \mid + \mid B \mid + \mid C \mid + \mid D \mid \\ \qquad\qquad - \delta(A,B) - \delta(A,C) \\ \qquad\qquad - \delta(B,D) - \delta(C,D)) \end{cases}$$

The first three equalities of the system can be visualized in the following figure:



2.2.2 Contiguity and Concatenation

An intuition about analogies is that two analogies could always be concatenated if they do not have any symbol in common. In fact, there are three possible ways of concatenating which meet intuition. However, formally, concatenating is not the only possibility. One could mix up symbols, for instance: $\varepsilon : a = bb : bab$. We shall reject this possibility.

AXIOM **3** *Let $\mathcal{V}$ be an alphabet, and $\mathcal{V}_1 \subset \mathcal{V}$, $\mathcal{V}_2 \subset \mathcal{V}$, such that $\mathcal{V}_1 \cap \mathcal{V}_2 = \emptyset$,* $\forall(A_1,B_1,C_1,D_1) \in ({\mathcal{V}_1}^*)^4$, $\forall(A_2,B_2,C_2,D_2) \in ({\mathcal{V}_2}^*)^4$ *such that,*

$$A_1 : B_1 = C_1 : D_1 \quad \textit{and} \quad A_2 : B_2 = C_2 : D_2$$

we postulate that there are only three possible ways of concatenating the analogies

$$\begin{cases} A_1A_2 : B_1B_2 = C_1C_2 : D_1D_2 \\ A_1A_2 : B_1B_2 = C_2C_1 : D_2D_1 \\ A_1A_2 : B_2B_1 = C_2C_1 : D_1D_2 \end{cases}$$

Thus, $\varepsilon : a = bb : bab$ is no more a valid analogy for us. The previous axiom implies that there are only two possible solutions to the analogical equation: $\varepsilon : a = bb : \mathtt{x} \quad \Rightarrow \quad \mathtt{x} = abb \vee \mathtt{x} = bba$, and consequently, the analogical equation $ab : aabb = aabb : \mathtt{x}$ has *aaabbb* as a unique

solution, and *aababb* is not a solution. The previous axiom is necessary in the proof of Theorems 4.1 and 4.2.

3 Formal Languages

Having studied analogies between strings of symbols, and having posited and deduced some formal results, we may now turn to the construction of sets of strings of symbols using analogies.

3.1 Immediate Analogical Derivation

We first introduce immediate analogical derivation.

Definition 3.1 Let $\mathcal{V}$ be an alphabet. The immediate analogical derivation, noted $\vdash_{\overline{\mathcal{M}}}$, modulo a set $\mathcal{M} \subset \mathcal{V}^* \times \mathcal{V}^*$, whose elements (v, v') are noted $v \to v'$, is defined in the following way:

$$\forall (w, w') \in \mathcal{V}^* \times \mathcal{V}^*, \quad w \vdash_{\overline{\mathcal{M}}} w' \quad \Leftrightarrow \quad \exists v \to v' \in \mathcal{M} \ / \ w : w' = v : v'$$

Although we use the notation $\to$ for the elements of $\mathcal{M}$, it is not to be interpreted in the way it would be in classical rewriting systems. This notation is just to make a parallel with the classical presentations of grammars, where the elements of $\mathcal{M}$ are called rules. However, the meaning here is different. With standard rules, w is exactly matched against v to produce, in a second step, w'. Here, the result w' depends on the way v (not w) "matches" w and v' at the same time.

3.2 Analogical Languages

We now show how some languages, *i.e.*, some sets of symbol strings, can be constructed with a device based on analogy.

Definition 3.2 Let $\mathcal{V}$ be an alphabet. Let $\mathcal{A} \subset \mathcal{V}^*$ and $\mathcal{M} \subset \mathcal{V}^* \times \mathcal{V}^*$, both finite. The language of analogical strings $\Lambda(\mathcal{A}, \mathcal{M})$ is defined in the following way:

$$\Lambda(\mathcal{A}, \mathcal{M}) = \mathcal{A} \cup \{ \ w' \in \mathcal{V}^* \mid \exists w \in \mathcal{A} \ / \ w \vdash^{+}_{\overline{\mathcal{M}}} w' \ \}$$

with $\vdash^{+}_{\overline{\mathcal{M}}}$, the transitive closure of the immediate analogical derivation $\vdash_{\overline{\mathcal{M}}}$.

The previous definition conforms to the usual presentation of formal languages. It aims at the generation of a language. Thus, as usual, standard structural induction is used to generate all of the members of a language of analogical strings. Starting with the elements of $\mathcal{A}$, all possible analogies with the elements of $\mathcal{M}$ as models are applied.

The reciprocal problem of generation is that of recognition. With an analogical system, the grammaticality of a given string, *i.e.*, its membership in a language, is tested against the set of attested strings of that language, after the reduction of that given string, by analogy, using the set of models. For recognition, the strings in the pairs of $\mathcal{M}$ are used in the reverse direction they appear in $\mathcal{M}$, and the analogies are solved in the direction reverse to that for generation. This is possible thanks to the inversion of ratios in Theorem 2.1.

The "linguistic" interpretation of a language of analogical strings $\Lambda(\mathcal{A},\mathcal{M})$ is thus as follows: $\mathcal{A}$ is the set of attested strings, *i.e.*, the set of strings against which any candidate element of the language will be compared *in fine*; $\mathcal{M}$ is the set of paradigmatic models (declensions, conjugations, morphological derivations, syntactic transformations, *etc.*), according to which any candidate element of the language is reduced [7] by analogy.

4 Where Are Analogical Languages?

We shall now briefly consider the relevance of analogical languages to the study of natural language from the point of view of the class of such languages.

4.1 Examples of Languages

It is possible to prove, by induction and with the use of Axiom 3, that the following three famous regular, context-free and, context-sensitive languages are all languages of analogical strings:

$$\{a^n \ / n \geq 1\} = \Lambda(\{a\},\{a \rightarrow aa\})$$
$$\{a^n b^n \ / n \geq 1\} = \Lambda(\{ab\},\{ab \rightarrow aabb\})$$
$$\{a^n b^n c^n \ / n \geq 1\} = \Lambda(\{abc\},\{abc \rightarrow aabbcc\})$$

and that, more generally:

Theorem 4.1
$\{a_1^n a_2^n \ldots a_m^n \ / \ n \geq 1\} = \Lambda(\{a_1 a_2 \ldots a_m\},\{(a_1 a_2 \ldots a_m \rightarrow a_1^2 a_2^2 \ldots a_m^2)\})$

Proof. For $\{a^n / n \geq 1\}$.

Completeness: $\Lambda(\{a\},\{a \rightarrow aa\}) \subset \{a^n / n \geq 1\}$. Recall that $\overline{w}$ is the set of different symbols in string w. Suppose that $w \in \Lambda(\{a\},\{a \rightarrow aa\})$. This is equivalent to: $a \overset{*}{\vdash\!\!-} w$. Hence, there exists a sequence of strings w_1, w_2,

[7] The word *reduce* is taken to mean a reduction to a normal form, not in the sense that the strings would become shorter.

$\ldots$, w_n such that the first column in the following array holds.

$$\begin{array}{ccccc}
a : w_1 = a : aa & \Leftrightarrow & w_1 : a = aa : a & \Rightarrow & \overline{w_1} \subset \overline{a} \cup \overline{aa} = \{a\} \\
w_1 : w_2 = a : aa & \Leftrightarrow & w_2 : w_1 = aa : a & \Rightarrow & \overline{w_2} \subset \overline{w_1} \cup \overline{aa} \\
\vdots & & \vdots & & \vdots \\
w_n : w = a : aa & \Leftrightarrow & w : w_n = aa : a & \Rightarrow & \overline{w} \subset \overline{w_n} \cup \overline{aa}
\end{array}$$

The second column in the array is obtained by inversion of ratios; the third column is the application of Axiom 1. This last column implies: $\overline{w} \subset \{a\}$, which means that w is of the type a^n.

Consistency: $\{a^n/n \geq 1\} \subset \Lambda(\{a\}, \{a \rightarrow aa\})$. By induction on n, any string of the form a^n is obtained by analogy with an element of $\Lambda(\{a\}, \{a \rightarrow aa\})$. Base: $\{a\} \subset \Lambda(\{a\}, \{a \rightarrow aa\})$ by the definition of a language of analogical strings. Induction: suppose that a^n is a member of $\Lambda(\{a\}, \{a \rightarrow aa\})$. The unique solution $\mathtt{x}$ of the analogical equation $a : aa = a^n : \mathtt{x}$ is $a^{n+1} \in \{a^n/n \geq 1\}$.

For $\{a^n b^n/n \geq 1\}$.

Completeness: $\Lambda(\{ab\}, \{ab \rightarrow aabb\}) \subset \{a^n b^n/n \geq 1\}$. A rationale similar to the one above gives $w \in \{a^n b^n/n \geq 1\} \Rightarrow \overline{w} \subset \{a, b\}$. By induction, by Axiom 3, all a's are before the b's, hence $w = a^n b^m$ with n necessarily equal to m.

Consistency: $\{a^n b^n/n \geq 1\} \subset \Lambda(\{ab\}, \{ab \rightarrow aabb\})$. By induction on n. Base: $ab \in \Lambda(\{ab\}, \{ab \rightarrow aabb\})$ is true, by definition of a language of analogical strings. Induction: suppose that $a^n b^n$ is a member of $\Lambda(\{ab\}, \{ab \rightarrow aabb\})$. Because $a : aa = a^n : a^{n+1}$ and $b : bb = b^n : b^{n+1}$ are true analogies, and by Axiom 3, the unique solution $\mathtt{x}$ of the analogical equation $ab : aabb = a^n b^n : \mathtt{x}$ is $a^{n+1} b^{n+1} \in \{a^n b^n/n \geq 1\}$.

For $\{a^n b^n c^n/n \geq 1\}$.

The proof is the same as for $\{a^n b^n/n \geq 1\}$, by decomposing

$$abc : aabbcc = a^n b^n c^n : a^{n+1} b^{n+1} c^{n+1}$$

into $ab : aabb = a^n b^n : a^{n+1} b^{n+1}$ and $c : cc = c^n : c^{n+1}$ which both hold.

Identical rationales prove that $\{a_1^n a_2^n \ldots a_m^n \ / \ n \geq 1\}$ is a language of analogical strings.

□

As a result, expressed as languages of analogical strings, the famous three examples for regular, context-free and, context-sensitive languages are very simple (one attested string and one derivation rule only), and, fortunately, absolutely similar.

Using a similar proof as for the theorem above, it is easy to prove that:

Theorem 4.2
$\{a^m b^n c^m d^n \ / \ n \geq 1 \wedge m \geq 1\} = \Lambda(\{abcd\}, \{abcd \rightarrow abbcdd, abcd \rightarrow aabccd\})$

This context-sensitive language, which is thus also a language of analogical strings, is the basis of two famous counter-examples against the context-freeness of natural language: in the morphology of Bambara [1], and in the syntax of the Zurich dialect of Swiss German [17]. It is worth noting that, expressed as a language of analogical strings, this language is quite simple.

Thus, some context-sensitive languages are languages of analogical strings. The question is: does that go too far?

4.2 Constant Growth

Mild context-sensitivity was proposed in [7] to characterize the family of languages that suits natural language (larger than context-free, but strictly smaller than context-sensitive), and a characterization in four properties was proposed in [8]. One of these properties, that of constant growth, is weaker than semilinearity ([12] refuted recently that natural languages would be semilinear) :

Definition 4.3 A language $\mathcal{L}$ has the constant growth property if (and only if) when arranging the strings of the language in increasing order of length, two consecutive lengths do not differ by arbitrarily large amounts.

Now, it can be proven using Lemma 2.3 that:

Theorem 4.4 *Any language of analogical strings verifies the constant growth property.*

Proof. Let $\Lambda(\mathcal{A},\mathcal{M})$ be a language of analogical strings. Let us call $k_{\mathcal{A}}$ the maximum over all $\mid w \mid$ with w in $\mathcal{A}$. $k_{\mathcal{A}}$ exists, because $\mathcal{A}$ is finite. Let us call $k_{\mathcal{M}}$ the maximum over all $\mid v' \mid - \mid v \mid$ with $v \rightarrow v'$ in $\mathcal{M}$. $k_{\mathcal{M}}$ exists also, because $\mathcal{M}$ is also finite.

$\Lambda(\mathcal{A},\mathcal{M})$ is generated by induction, starting with the elements of $\mathcal{A}$, by application of immediate analogical derivations with the elements of $\mathcal{M}$.

At the beginning of the generation, the set $\mathcal{A}$ has the constant growth property, with $k_{\mathcal{A}}$, a possible (too large) bound for the amounts between two consecutive lengths.

Suppose that, at one step of the generation by induction, the generated set of all strings created until that step has the constant growth property.

In the next step, a new string is generated with the help of an element w obtained during the last step. In fact, zero, one, or several [8] elements w' may be generated with the help of any element $v \rightarrow v'$ of $\mathcal{M}$, depending whether the analogy $v : v' = w : \mathtt{x}$ has zero, one, or several solutions. When there exists at least one such solution w', Lemma 2.3 tells us that $\mid v \mid + \mid w' \mid = \mid v' \mid + \mid w \mid$, thus $\mid w' \mid - \mid w \mid = \mid v' \mid - \mid v \mid$. If $\mid w' \mid - \mid w \mid \leq 0$, the length decreases, and $\mid w' \mid$ can be arranged in between lengths of

[8] Axiom 1 and Lemma 2.3 imply that an analogical equation has a finite number of solutions.

already generated elements of the language, and even possibly before $k_{\mathcal{A}}$. If $\mid w' \mid - \mid w \mid > 0$, the length increases, but by less than $k_{\mathcal{M}}$. As this is true for all new strings w' generated during the new step, consequently, the new set of generated strings union the set of strings generated until that step has the constant growth property, with $k = \max(k_{\mathcal{A}}, k_{\mathcal{M}})$ as a bound for the amounts between two consecutive string lengths arranged in increasing order.

This concludes the proof by induction. □

Consequently, a language like $\{a^{2^n}/n \in \mathbb{N}\}$ is not a language of analogical strings, as it does not have the bounded growth property. Luckily thus, some "unnatural" languages are out of the reach of languages of analogical strings.

Conclusion

Only a small number of proposals have been made for the modelization of analogy, maybe because the dominant stream in linguistics for years, the generative one, against works by the founders of modern linguistics (*e.g.*, [13, chap. V & XII] or [15, Part III, Chap. 4 & 5]), explicitly rebutted analogy as a possible object of research (see [6, 132 and 136], for quotations from Chomsky). However, according to other arguments [5], analogy may be argued to be a component in language (of course, surely not the only one).

We have shown how to generate a family of formal languages, called languages of analogical strings. It is important to note that their construction, as is the case with simple contextual grammars [4], does not make any use of non-terminals. Grammaticality is simply tested against some attested strings, after reduction according to some models. The approach by reduction to attested forms has already been advocated in natural language processing [14].

The key language $\{a^n b^m c^n d^m / n \geq 1\}$ against the context-freeness hypothesis of natural language is easily shown to be a language of analogical strings. Also, all languages of analogical strings possess the constant growth property, which intervenes partially in mild context-sensitivity, a notion introduced to cope with the apparent power of human languages.

References

[1] CULY, C., *The complexity of the vocabulary of Bambara*, Linguistics and Philosophy **8** (1985), pp. 345–351.

[2] DOWTY, D., L. KARTTUNEN and A. ZWICKY, editors, "Natural language processing," Cambridge University Press, Cambridge, 1985.

[3] GENTNER, D., *Structure mapping: A theoretical model for analogy*, Cognitive Science **7** (1983), pp. 155–170.

[4] ILIE, L., "On Ambiguity in Internal Contextual Languages," in [11], 1998, pp. 29–45.

[5] ITKONEN, E., *Iconicity, analogy, and universal grammar*, Journal of Pragmatics **22** (1994), pp. 37–53.

[6] ITKONEN, E. and J. HAUKIOJA, "A rehabilitation of analogy in syntax (and elsewhere)," in [9], 1997, pp. 131–177.

[7] JOSHI, A., "Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural description?" in [2], 1985, pp. 206–250.

[8] JOSHI, A., K. VIJAY-SHANKER and D. WEIR, "The Convergence of Mildly Context-Sensitive Grammar Formalisms," in [16], 1991, pp. 31–81.

[9] KERTÉSZ, A., "Metalinguistik im Wandel: die kognitive Wende in Wissenschaftstheorie und Linguistik," Peter Lang, Frankfurt a/M, 1997.

[10] LEVENSHTEIN, V., *Binary codes capable of correcting deletions, insertions and reversals*, Soviet Physics-Doklady **10** (1966), pp. 707–710.

[11] MARTÍN-VIDE, C., editor, "Mathematical and computational analysis of natural language," John Benjamins Publishing Co., Amsterdam / Philadelphia, 1998.

[12] MICHAELIS, J. and M. KRACHT, "Logical Aspects of Computational Linguistics," Number 1328 in LNCS/LNAI, Springer Verlag, Berlin, 1997, 329–345 pp.

[13] PAUL, H., "Prinzipien der Sprachgeschichte," Niemayer, Tübingen, 1920 [5th ed., 1st ed. 1880].
http://www.gutenberg.aol.de/paulh/prinzip/paulvorr.htm

[14] SAGER, N., "Natural Language Information Processing: A Computer Grammar of English and Its Applications," Addison-Wesley, Reading, Mass., 1981.

[15] SAUSSURE, F., "Cours de linguistique générale," Payot, Lausanne et Paris, 1995 [1st ed. 1916].

[16] SELLS, P., S. SHIEBER and T. WASOW, editors, "Foundational Issues in natural language processing," MIT Press, Cambridge, 1991.

[17] SHIEBER, S. M., *Evidence against the context-freeness of natural language*, Linguistics and Philosophy **8** (1985), pp. 333–343.

[18] STEPHEN, G. A., "String searching algorithms," World Scientific, Singapore–New Jersey–London–Hong Kong, 1998.