LANGUAGES OF ANALOGICAL STRINGS

Yves Lepage
ATR Spoken Language Translation Research Labs,
Hikari-dai 2-2-2, Seika-tyō, Sōraku-gun, Kyōto 619-0288, Japan
yves.lepage@slt.atr.co.jp

1 Introduction

Analogies between strings of symbols, noted[1] $A : B = C : D$, put four strings of symbols into "proportions." They render an account of, for instance, *look* : *looked* = *walk* : *walked* or *fable* : *fabulous* = *miracle* : *miraculous*, on the level of strings of symbols. They are not intended to deal directly with, for instance, *bird* : *wings* = *fish* : *fins* or *work* : *worked* = *go* : *went* which suppose knowledge about the world or the tongue (Hoffman 95). Analogies may be read as equalities, as well as equations to be solved, as in:

to look : *I looked* = *to act* : x $\Rightarrow$ x = *I acted*

The goal of this paper is to establish some fundamental, common-sense hypotheses (axioms) about analogies in general; then to draw from them basic results (theorems) on analogies between strings of symbols in particular; so as to propose a possible definition for languages of analogical strings; and to prove that some famous languages of particular interest to the language processing community are very simple languages in this respect. We further argue that the fact that the property of bounded growth is verified by any such language is in favour of modelling part of natural language using such languages.

Our feeling is that analogy between strings of symbols is an operation as fundamental as, *e.g.*, addition is to natural numbers. However, to our knowledge, letting aside the Copycat project (Hofstadter et al. 94, Chap. 5–7, pp. 195–318) which has no such goals and relies on different methods, no mathematical formalisation has ever been proposed for analogies between strings of symbols.

[1] In the sequel, A, B, C and D are *variables* denoting objects.

2 General Properties of Analogy

We start with results which hold independently of the set to which the terms of the analogy belong.

2.1 Fundamental Hypotheses

In the Nicomachean Ethics (Book V), Aristotle wrote:

> For proportion is equality of ratios, and involves four terms at least [...] As the term A, then, is to B, so will C be to D, and therefore, alternando, as A is to C, B will be to D. [Translation by W. D. Ross]

As a consequence, we shall hypothesize the following property:

Axiom 1 (Exchange of the means)
$A : B = C : D \Leftrightarrow A : C = B : D$

Another equivalence is also used by Aristotle in his Poetics. It is based on the symmetry of the equality (the word "as," here): if we can say that A is to B as C is to D, then we should also be able to say that C is to D as A is to B.

Axiom 2 (Symmetry of equality)
$A : B = C : D \Leftrightarrow C : D = A : B$

2.2 Equivalent Forms of Analogy

By successive application of the previous hypotheses, we get eight equivalent forms of the same analogy, listed hereafter in the alphabet order of the term variables A, B, C and D.

Theorem 1 (Equivalent forms) *The eight following analogies are equivalent:*

$A : B = C : D$ *(i)*
$A : C = B : D$ *(ii)*
$B : A = D : C$ *(iii)* $\Leftarrow$ *ii+vi+ii*
$B : D = A : C$ *(iv)* $\Leftarrow$ *iii+ii*
$C : A = D : B$ *(v)* $\Leftarrow$ *ii+iii*
$C : D = A : B$ *(vi)*
$D : B = C : A$ *(vii)* $\Leftarrow$ *ii+vi+iii*
$D : C = B : A$ *(viii)* $\Leftarrow$ *iii+vi*

Some interesting results may be obtained on the number of different possible analogy classes given four objects. However, we shall leave them aside for lack of space.

3 Analogy on Strings of Symbols

We shall now specialise on the case where the members of the analogy being considered belong to a set of strings of symbols. The structure of strings implies new properties.

3.1 Examples

In order to support the next hypothesis we will make on analogies on strings of symbols, let us list a small number of analogies in English:

hypothesis : *hypotheses* = *thesis* : *theses*
leaf : *leaves* = *calf* : *calves*
give : *gave* = *sing* : *sang*
inexact : *exact* = *incapable* : *capable*

plus some true analogies but with no meaning in language:

$aa : aaaa = aaaa : aaaaaa$ [2]
give : *gave* = *bid* : *bad*
walk : *walked* = *go* : *goed* [3]

and some counter-examples (noted with $\neq$):

$aaaa : bbbb \neq cccc : dddd$ [4]
dfhka : *bzvmbz* $\neq$ *bzvmbz* : *dfhka*

[2] This analogy holds independently of the truth (or falsity of) $aa : aaaa = aaaa : aaaaaaaa$ ($a^2 : a^4 = a^4 : a^8$). In fact, hypothesising $A : B = AA : BB$ for any string A and B is incompatible with the Symbol inclusion axiom because the Equality of length sums on $a^n : a^m = a^{2n} : a^{2m}$ would yield $n + 2m = m + 2n$, *i.e* $n = m$, for any n, $m \in \mathbb{N}$, which is absurd.

[3] Refrain from thinking in English, and recall that we work on the sole level of symbols: *i* just became *a*, or *ed* has just been added.

[4] In absence of any knowledge about the world. Here, only the equality between symbols can be tested. Because the alphabetical order is not known, this analogy cannot be verified.

3.2 Symbol Inclusion

By inspection of the previous examples, one can state that there is no solution to an analogy on the strings of symbols $A : B = C : \mathtt{x}$ if some symbols of A appear neither in B nor in C. The contrapositive is that, for an analogy to hold, any symbol of A has to appear in either B or C. Noting by $\overline{A}$ the set of symbols contained in A, we restate the previous observation as the following hypothesis which will be used in Appendix in the proofs that some well-known languages are languages of analogical strings (Theorems 5 and 6 of Section 5.1).

Axiom 3 (Symbol inclusion) *Let $\mathcal{V}$ be an alphabet.* $\forall (A,B,C,D) \in (\mathcal{V}^*)^4$,

$$A : B = C : D \quad \Rightarrow \quad \overline{A} \subset \overline{B} \cup \overline{C}$$

For strings reduced to one symbol, this trivially implies: $a : b = b : a \Leftrightarrow a = b$.

Incidently, applied on the eight equivalent forms of an analogy, the Symbol inclusion axiom implies eight inclusions, of which, only four are distinct by commutativity of union. These four inclusions imply, and are implied by, two reciprocal inclusions:

$$\begin{cases} \overline{A} \subset \overline{B} \cup \overline{C} \\ \overline{B} \subset \overline{A} \cup \overline{D} \\ \overline{C} \subset \overline{A} \cup \overline{D} \\ \overline{D} \subset \overline{B} \cup \overline{C} \end{cases} \Leftrightarrow \begin{cases} \overline{A} \cup \overline{D} \subset \overline{B} \cup \overline{C} \\ \overline{B} \cup \overline{C} \subset \overline{A} \cup \overline{D} \end{cases}$$

so that, one can state:

Theorem 2
Let $\mathcal{V}$ be an alphabet. $\forall (A,B,C,D) \in (\mathcal{V}^*)^4$,

$$A : B = C : D \quad \Rightarrow \quad \overline{A} \cup \overline{D} = \overline{B} \cup \overline{C}$$

3.3 Similarity Constraint

The Symbol inclusion axiom can be refined by saying that, the sum of the similarities[5] of A with B and C must be greater than or equal to its length: $\mathrm{sim}(A,B) + \mathrm{sim}(A,C) \geq |A|$

When the length of A is less than the sum of the similarities, some symbols of A are common

[5] The *similarity* between two strings is defined as the length of their longest common subsequence (Hirschberg 75). A *subsequence* of a string is any not necessarily connex sequence of symbols from that string in the same order.

to all strings, A, B, and C in the same order, and these symbols are necessarily present in D in the same order also. We call $\gamma(A,B,C,D)$ the number of such symbols. As a result, $\forall(A,B,C,D) \in (\mathcal{V}^*)^4,\ A:B=C:D \ \Rightarrow$
$|A| = \mathrm{sim}(A,B) + \mathrm{sim}(A,C) - \gamma(A,B,C,D)$

The Equivalent forms theorem yields:

$$
\begin{array}{l}
|A| = \mathrm{sim}(A,C) + \mathrm{sim}(A,B) - \gamma(A,C,B,D) \\
|B| = \mathrm{sim}(B,A) + \mathrm{sim}(B,D) - \gamma(B,A,D,C) \\
|B| = \mathrm{sim}(B,D) + \mathrm{sim}(B,A) - \gamma(B,D,A,C) \\
|C| = \mathrm{sim}(C,A) + \mathrm{sim}(C,D) - \gamma(C,A,B,D) \\
|C| = \mathrm{sim}(C,D) + \mathrm{sim}(C,A) - \gamma(C,D,A,B) \\
|D| = \mathrm{sim}(D,B) + \mathrm{sim}(D,C) - \gamma(D,B,C,A) \\
|D| = \mathrm{sim}(D,C) + \mathrm{sim}(D,B) - \gamma(D,C,B,A)
\end{array}
$$

Because all $\gamma(.,.,.,.)$ are equal in all the equalities above, and by the symmetry of similarity, the substraction of pairs of lines yields the following theorem, which is necessary for the proof of our theorem on bounded growth property (Theorem 7 of Section 5.2).

Theorem 3 (Similarity constraint)
Let $\mathcal{V}$ be an alphabet. $\forall(A,B,C,D) \in (\mathcal{V}^*)^4$,
$A:B=C:D \ \Rightarrow$

$$
\left\{
\begin{array}{rcl}
|A| - \mathrm{sim}(A,B) & = & |C| - \mathrm{sim}(C,D) \\
|B| - \mathrm{sim}(B,D) & = & |A| - \mathrm{sim}(A,C) \\
|C| - \mathrm{sim}(C,A) & = & |D| - \mathrm{sim}(D,B) \\
|D| - \mathrm{sim}(D,C) & = & |B| - \mathrm{sim}(B,A)
\end{array}
\right.
$$

3.4 Equality of length sums

A remarkable theorem is easily derived from the Similarity constraint theorem by addition and substraction and by commutatitivity of similarity.

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

$$A:B=C:D \ \Rightarrow \ |A| + |D| = |B| + |C|$$

3.5 Disjoint Analogies

Another intuitive idea about analogies between strings of symbols is that two analogies could always be concatenated. Whether this is true remains an open problem.

However, the previous intuition seems to hold anyway when the two analogies to be concatenated do not have any symbol in common. We call such analogies, *disjoint analogies*. The intuition is that, disjoint analogies may be applied one after another without any problem. But concatenating in the same order is not the only possibility.

One gets $2^4 = 16$ analogies by enumerating all possibilities of exchanging or not exchanging the substrings indexed by 1 and 2 in $A_1A_2 : B_1B_2 = C_1C_2 : D_1D_2$. By numbering these 16 analogies using a binary notation reflecting the place where this exchange took place, numbers which are binary complements denote two equivalent analogies, of which one may be eliminated from the list. We list hereafter those analogies with A_1A_2 as a first term.

$$
\begin{array}{ll}
(0000) & A_1A_2 : B_1B_2 = C_1C_2 : D_1D_2 \\
(0001) & A_1A_2 : B_1B_2 = C_2C_1 : D_2D_1 \\
(0010) & A_1A_2 : B_1B_2 = C_2C_1 : D_1D_2 \\
(0011) & A_1A_2 : B_1B_2 = C_2C_1 : D_2D_1 \\
(0100) & A_1A_2 : B_2B_1 = C_1C_2 : D_1D_2 \\
(0101) & A_1A_2 : B_2B_1 = C_1C_2 : D_2D_1 \\
(0110) & A_1A_2 : B_2B_1 = C_2C_1 : D_1D_2 \\
(0111) & A_1A_2 : B_2B_1 = C_2C_1 : D_2D_1
\end{array}
$$

The number of different cases is further reduced using $(i) \Leftrightarrow (viii)$ of the Equivalent forms: $(0001) \Leftrightarrow (1000) \Leftrightarrow (0111)$ and $(0010) \Leftrightarrow (0100)$. The reduced set is: $\{ (0000),(0001),(0010),(0011),(0101),(0110) \}$.

Similarly, $(i) \Leftrightarrow (ii)$ of the Equivalent forms yields the equivalences: $(0010) \Leftrightarrow (0100)$ and $(0011) \Leftrightarrow (0101)$. The reduced set becomes: $\{ (0000),(0001),(0011),(0110) \}$.

Of these four possible analogies, the second one, (0001), where only one exchange is performed, is not true in general. For instance, $ay : az = by : \mathtt{x}$ is not acceptable when $\mathtt{x} = zb$. On the contrary, the three other possible analogies meet intuition, so that the following hypothesis may be laid.

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

$$
\left.
\begin{array}{l}
A_1 : B_1 = C_1 : D_1 \\
A_2 : B_2 = C_2 : D_2
\end{array}
\right\} \Rightarrow
\left\{
\begin{array}{l}
A_1A_2 : B_1B_2 = C_1C_2 : D_1D_2 \\
A_1A_2 : B_1B_2 = C_2C_1 : D_2D_1 \\
A_1A_2 : B_2B_1 = C_2C_1 : D_1D_2
\end{array}
\right.
$$

This axiom will be used in Appendix in the proof of Theorems 5 and 6 of Section 5.1.

4 Languages of analogical strings

4.1 Analogical Derivation

In order to show how some languages, *i.e.*, some sets of symbol strings, can be characterised by a device based on analogy, we first introduce *analogical derivations*. We intentionally use this term to make a parallel with the vocabulary of formal grammars.

Definition 1 *Let $\mathcal{V}$ be an alphabet. The analogical derivation, noted $\vdash_{\overline{\mathcal{M}}}$, modulo a set $\mathcal{M} \subset \mathcal{V}^* \times \mathcal{V}^*$, whose elements (v, v') are noted $v \rightarrow v'$, is defined in the following way:*
$\forall (w, w') \in \mathcal{V}^* \times \mathcal{V}^*$,

$$w \vdash_{\overline{\mathcal{M}}} w' \quad \Leftrightarrow \quad \exists v \rightarrow v' \in \mathcal{M} \; / \quad w : w' = v : v'$$

Although we use the notation $\rightarrow$ 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 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.*

4.2 Derivational Systems

Definition 2 *A derivational system of analogical strings is a triple $G = (\mathcal{V}, \mathcal{A}, \mathcal{M})$, where $\mathcal{V}$ is a finite alphabet, $\mathcal{A} \subset \mathcal{V}^*$ (finite) is the set of axioms, or, better, the set of* attested strings, *and $\mathcal{M} \subset \mathcal{V}^* \times \mathcal{V}^*$ (finite) is the set of rules, or, better, the set of* models.

4.3 Languages

Definition 3 *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 $\Lambda(\mathcal{A}, \mathcal{M}) = < \mathcal{A}, \{\vdash_{\overline{\mathcal{M}}}\} >$ is defined in the following way:*

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

with $\overset{+}{\vdash}_{\overline{\mathcal{M}}}$, the transitive closure of the 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 order they appear in $\mathcal{M}$, and the analogies are solved in the other direction than for generation. This is possible thanks to form (iii) of the Equivalent forms theorem.

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[6] by analogy.

5 Some Properties

5.1 $\{a_1^n a_2^n \ldots a_m^n\}$ and $\{a^m b^n c^m d^n\}$

In appendix, we give proofs that the following famous regular, context-free and, context-sensitive languages are all languages of analogical strings:

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

and that, more generally:

Theorem 5 $\{a_1^n a_2^n \ldots a_m^n \; / \; n \geq 1\} = \Lambda(\{a_1 a_2 \ldots a_m\}, \{(a_1 a_2 \ldots a_m \rightarrow a_1^2 a_2^2 \ldots a_m^2)\})$

In a similar way, by induction and use of the Concatenation of disjoint analogies, it is easy to prove that:

Theorem 6 $\{a^m b^n c^m d^n \; / \; n \geq 1 \wedge m \geq 1\} = \Lambda(\{abcd\}, \{abcd \rightarrow abbcdd, abcd \rightarrow aabccd\})$

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

This language is famous for being the basis of two counter-examples against the context-freeness of natural language: in the morphology of Bambara (Culy 85), and in the syntax of the Zurich dialect of Swiss German (Shieber 85).

5.2 Bounded Growth

Following the discussion about the non-context-freeness of natural language, the family of formal languages that can be used to formalise natural language has been thought to be necessarily larger than the family of context-free languages, but it does not have to cover all context-sensitive languages, as some context-sensitive languages are obviously not relevant for natural languages. *Mild context-sensitivity* was thus proposed by (Joshi 85) to characterise the family of languages captured by tree-adjoining grammars (larger than context-free, but strictly smaller than context-sensitive).

However, this is a characterisation by a recognition device, and some have proposed other intrinsic characterisations. (Marcus & al. 96) have been advocating that, the key point in "mild context-sensitivity" is the property of bounded growth: for each sentence in a language, we can always find another sentence in the same language whose length differs from the length of the first sentence by at most a given constant.

Definition 4 (Bounded growth) *A language $\mathcal{L}$ has the bounded growth property if (and only if) $\mathcal{L}$ is a singleton or $\exists k \in \mathbb{N}$ /*

$$\forall w \in \mathcal{L},\ \exists w' \in \mathcal{L} \backslash \{w\} \quad / \quad \Big||w'| - |w|\Big| \leq k$$

Now, it is easy to prove (see Appendix) that:

Theorem 7 *Any language of analogical strings verifies the bounded growth property.*

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.

6 Conclusion

Only a small number of proposals have been made for the modelisation of analogy, the rare exceptions being (Itkonen & Haukioja 97) and, out of linguistics, (Hofstadter et al. 94), maybe because the dominant stream in linguistics for years, the generative one, against works by the founders of modern linguistics (*e.g.* (Saussure 16, Part III, Chap. 4 & 5)), explicitly rebutted analogy as a possible object of research (see (Itkonen & Haukioja 97, 132 and 136), for quotations from Chomsky) under the fallacious pretext that blind application of analogy may lead to falsity in logic and agrammaticality in syntax. However, following recent results in experimental psychology and refutations of the innateness hypothesis (Itkonen 94), analogy may reasonably be argued to be a component in language (of course, surely not the only one).

Having posited only four fundamental hypotheses on analogy, we have shown how to generate a family of formal languages, called languages of analogical strings. It is important to note that analogical string grammars, like simple contextual grammars (Ilie 96), do 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 (Sager 81).

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 bounded growth property, which attempts to capture *mild context-sensitivity*, a notion introduced to cope with the apparent power of human languages.

The fact that the regular language $\{a^n\}$, the context-free language $\{a^n b^n\}$, and the context-sensitive language $\{a^n b^n c^n\}$ are very similar languages of analogical strings shows that analogy allows us "to get round" the Chomsky classification.

7 Acknowledgements

Thanks to Pr. Boitet for fruitful discussions about the content of this paper.

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

Appendix

Theorems 5 and 6

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

Completion: $\Lambda(\{a\},\{a \rightarrow aa\}) \subset \{a^n/n \geq 1\}$. 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: $a \stackrel{*}{\vdash\!\!-} w$. Hence, there exists a sequence of strings w_1, w_2, ..., w_n such that the first column in the set of relations at the top of this page holds; the second column is the equivalent form (iii); the third column is the application of the Symbol inclusion axiom. This last column implies: $\overline{w} \subset \{a\}$, which means that w is of the type a^n (note that there is no empty string here).

Consistence: $\{a^n/n \geq 1\} \subset \Lambda(\{a\},\{a \rightarrow aa\})$. 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 solution $\mathtt{x}$ of the analogy $a^n : \mathtt{x} = a : aa$ is $a^{n+1} \in \{a^n/n \geq 1\}$. **QED**.

Proof: for $\{a^n b^n/n \geq 1\}$.

Completion: $\Lambda(\{ab\},\{ab \rightarrow aabb\}) \subset \{a^n b^n/n \geq 1\}$. 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 the Concatenation of disjoint analogies, all a's are before the b's, hence $w = a^n b^m$ with n necessarily equal to m.

Consistence: $\{a^n b^n/n \geq 1\} \subset \Lambda(\{ab\},\{ab \rightarrow aabb\})$. 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 $a^n : a^{n-1} = aa : a$ and $b^n : b^{n-1} = bb : b$ are true analogies, and by the Concatenation of disjoint analogies axiom, the solution $\mathtt{x}$ of the analogy $a^n b^n : \mathtt{x} = ab : aabb$ is $a^{n+1} b^{n+1} \in \{a^n b^n/n \geq 1\}$. **QED**.

Proof: 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

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

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

Identical rationales prove Theorems 5 and 6.

Theorem 7

Proof: Let $\Lambda(\mathcal{A},\mathcal{M})$ be a language of analogical strings not reduced to a singleton. For a given w in this language, either w is in $\mathcal{A}$ or not. In the first case, as $\mathcal{A}$ is finite, $k_{\mathcal{A}}$ the maximum over all $\Big||w'| - |w|\Big|$ with w and w' in $\mathcal{A}$, exists. In the case w is not in $\mathcal{A}$, by definition, there exists another element w' in the same language, and there exists $v' \rightarrow v \in \mathcal{M}$ such that $w' : w = v' : v$. The Similarity constraint implies:

$$\left\{ \begin{array}{l} |w'| - \mathrm{sim}(w',w) = |v'| - \mathrm{sim}(v',v) \\ |w| - \mathrm{sim}(w',w) = |v| - \mathrm{sim}(v',v) \end{array} \right.$$

Because $\mathrm{sim}(w,w') \leq \min(|w|,|w'|)$ is always true: $\Big||w'| - |w|\Big| = \max(|w|,|w'|) - \min(|w|,|w'|) \leq \max(|w|,|w'|) - \mathrm{sim}(w,w')$. Thus: $\forall w \in \Lambda(\mathcal{A},\mathcal{M}), \exists w' \in \Lambda(\mathcal{A},\mathcal{M}) \backslash \{w\}$ /

$$\begin{array}{rcl} \Big||w'| - |w|\Big| & \leq & \max(|w'| - \mathrm{sim}(w,w'), \\ & & \quad |w| - \mathrm{sim}(w,w')) \\ & \leq & \max(|v'| - \mathrm{sim}(v,v'), \\ & & \quad |v| - \mathrm{sim}(v,v')) \end{array}$$

By taking k the maximum over all $v \rightarrow v'$ of $\mathcal{M}$ and $k_{\mathcal{A}}$:

$$\forall w \in \Lambda(\mathcal{A},\mathcal{M}), \exists w' \in \Lambda(\mathcal{A},\mathcal{M}) \backslash \{w\} \ / \ \Big||w'| - |w|\Big| \leq k$$

QED.

References

Christopher Culy
The Complexity of the Vocabulary of Bambara
Linguistics and Philosophy, vol. 8, 1985, pp. 345-351.

Douglas Hofstadter and the Fluid Analogies Research Group
Fluid Concepts and Creative Analogies
Basic Books, New-York, 1994.

Robert R. Hoffman
Monster Analogies
AI Magazine, Fall 1995, vol. 11, pp 11-35.

Lucian Ilie
On Ambiguity in Internal Contextual Languages
in Carlos Martín-Vide (ed.), *Mathematical and computational analysis of natural language*, John Benjamins Publishing Co., Amsterdam / Philadelphia, 1998.

Esa Itkonen & Jussi Haukioja
A rehabilitation of analogy in syntax (and elsewhere)
in András Kertész (ed.) *Metalinguistik im Wandel: die kognitive Wende in Wissenschaftstheorie und Linguistik* Frankfurt a/M, Peter Lang, 1997, pp. 131-177.

Esa Itkonen
Iconicity, analogy, and universal grammar
Journal of Pragmatics, 1994, vol. 22, pp. 37-53.

Aravind K. Joshi
Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural description?
in Dowty *et al.*, *Natural language processing*, Cambridge University Press, Cambridge, 1985, pp 206-250.

D.S. Hirschberg
Algorithms for the longest common subsequence problem
Journal of the ACM, Vol. 24, No. 4, Oct. 1977, pp. 664-675.

Solomon Marcus, Carlos Martin-Vide, Gheorghe Păun
Contextual Grammars versus Natural Languages
Turku center for Computer Science, TUCS technical Report No 44, Sept. 1996.

Naomi Sager
Natural Language information Processing: A Computer Grammar of English and Its Applications
Addison-Wesley, Reading, Mass., 1981.

Ferdinand de Saussure
Cours de linguistique générale
publié par Charles Bally et Albert Sechehaye, Payot, Lausanne et Paris, 1916.

Stuart M. Shieber
Evidence against the Context-Freeness of Natural Language
Linguistics and Philosophy, vol. 8, 1985, pp. 333-343.