Unsupervised Morphological Analysis by Formal Analogy

Jean-François Lavallée and Philippe Langlais

DIRO, Université de Montréal
C.P. 6128, succursale Centre-ville
Montréal, Canada H3C 3J7
{lavalljf,felipe}@iro.umontreal.ca

Abstract. While classical approaches to unsupervised morphology acquisition often rely on metrics based on information theory for identifying morphemes, we describe a novel approach relying on the notion of *formal analogy*. A formal analogy is a relation between four forms, such as: *reader* is to *doer* as *reading* is to *doing*. Our assumption is that formal analogies identify pairs of morphologically related words. We first describe an approach which simply identifies all the formal analogies involving words in a lexicon. Despite its promising results, this approach is computationally too expensive. Therefore, we designed a more practical system which learns morphological structures using only a (small) subset of all formal analogies. We tested those two approaches on the five languages used in Morpho Challenge2009.

1 Introduction

Two major approaches are typically investigated for accomplishing unsupervised morphological analysis. The first one uses statistics in order to identify the most likely segmentation of a word. The basic idea is that low predicability of the upcoming letter in a string indicates a morpheme boundary. This approach has been around for quite some time. Indeed, Harris [1] described such a system in the fifties. Variants of this idea have recently been investigated as well. For instance, both the system in [2] as well as *Morfessor* [3] utilize perplexity as one feature to score potential segmentations. The second approach consists of grouping words into paradigms and removing common affixes . Variants of this approach [4, 5] have yielded very good results in Morpho Challenge 2008 and 2009 [6].

The potential of *analogical learning* in solving a number of canonical problems in computational linguistics has been the subject of recent research [7–9]. In particular, several authors have shown that analogical learning can be used to accomplish morphological analysis. Stroppa & Yvon [10] demonstrate its usefulness in recovering a word's lemma. They report state-of-the-art results for three languages (English, Dutch and German). Hathout [11, 12] reports an approach where morphological families are automatically extracted thanks to formal analogies and some semantic resources. However, to the best of our knowledge, it has

not been shown that analogical learning on a lexicon alone can be used as a means of acquiring a given language's morphology. This study aims to fill this gap.

The remainder of this paper is as follows. First, we provide our definition of formal analogy in Sect. 2. We then describe the two systems we devised based on this definition in Sect. 3. We present our experimental protocol and the results we obtained in Sect. 4. We conclude and discuss future avenues in Sect. 5.

2 Formal Analogy

A *proportional analogy*, or analogy for short, is a relation between four items noted $[\mathtt{x} : \mathtt{y} = \mathtt{z} : \mathtt{t}]$ which reads "$\mathtt{x}$ is to $\mathtt{y}$ as $\mathtt{z}$ is to $\mathtt{t}$". Among proportional analogies, we distinguish formal analogies, that is, those we can identify at a graphemic level, such as [*cordially* : *cordial* = *appreciatively* : *appreciative*].

Formal analogies can be specified in many ways [13] [14]. In this study we define them in terms of factorization. Let $\mathtt{x}$ be a string over alphabet Σ, a *n-factorization* of $\mathtt{x}$, noted $f_{\mathtt{x}}$, is a sequence of n *factors* $f_{\mathtt{x}} = (f^1_{\mathtt{x}}, \ldots, f^n_{\mathtt{x}})$, such that $\mathtt{x} = f^1_{\mathtt{x}} \odot f^2_{\mathtt{x}} \odot f^n_{\mathtt{x}}$, where $\odot$ denotes the concatenation operator. Based on [14] we therefore define a formal analogy as:

Definition 1. $\forall(\mathtt{x}, \mathtt{y}, \mathtt{z}, \mathtt{t}) \in \Sigma^{\star^4}$, $[\mathtt{x} : \mathtt{y} = \mathtt{z} : \mathtt{t}]$ ***iff*** *there exist d-factorizations* $(f_{\mathtt{x}}, f_{\mathtt{y}}, f_{\mathtt{z}}, f_{\mathtt{t}}) \in (\Sigma^{\star^d})^4$ *of* $(\mathtt{x}, \mathtt{y}, \mathtt{z}, \mathtt{t})$ *such that:* $\forall i \in [1, d]$, $(f^i_{\mathtt{y}}, f^i_{\mathtt{z}}) \in \{(f^i_{\mathtt{x}}, f^i_{\mathtt{t}}), (f^i_{\mathtt{t}}, f^i_{\mathtt{x}})\}$. *The smallest d for which this definition holds is called the degree of the analogy.*

According to this definition, [*cordially* : *cordial* = *appreciatively* : *appreciative*] is an analogy because we can find a quadruplet of 4-factorizations (factorizations involving 4 factors) as shown in the first column of Fig. 1. The second column of this figure also shows that a quadruplet of 2-factorizations also satisfies the definition. This illustrates the *alternations* passively captured by this analogy, that is, *appreciative*/*cordial* and *ly*/ϵ; the latter one (passively) capturing the fact that in English, an adverb can be constructed by appending *ly* to an adjective.

$f_{cordially}$	$\equiv$	*cordia*	*l l y*	$f_{cordially}$	$\equiv$	*cordial*	*ly*
$f_{cordial}$	$\equiv$	*cordia*	ϵ *l* ϵ	$f_{cordial}$	$\equiv$	*cordial*	ϵ
$f_{appreciatively}$	$\equiv$	*appreciative*	*l* ϵ *y*	$f_{appreciatively}$	$\equiv$	*appreciative*	*ly*
$f_{appreciative}$	$\equiv$	*appreciative*	ϵ ϵ ϵ	$f_{appreciative}$	$\equiv$	*appreciative*	ϵ

Fig. 1. Two factorizations of the analogy of degree 2 [*cordially* : *cordial* = *appreciatively* : *appreciative*].

3 Analogical Systems

The two systems we have designed rely on the assumption that a formal analogy implicitly identifies two pairs of forms that are morphologically related. For instance, the analogy in Fig. 1 relates *cordial* to *cordially*, as well as *appreciative* to *appreciatively*. Linking related words together is precisely the main task evaluated at Morpho Challenge. Therefore, given a lexicon $\mathcal{L}$, we need to identify all formal analogies involving its words. The following is the definition we use for such formal analogies:

$$\mathcal{A}(\mathcal{L}) = \{(x, y, z, t) \in \mathcal{L}^4 \ : \ [x : y = z : t]\}$$

Stroppa [15] describes a dynamic programming algorithm which checks whether a quadruplet of forms (x, y, z, t) is a formal analogy according to the previous definition. The complexity of this algorithm is in $O(|x| \times |y| \times |z| \times |t|)$.

As simple as it seems, identifying formal analogies is a very time consuming process. A straightforward implementation requires checking $O(|\mathcal{L}|^4)$ analogies, where $|\mathcal{L}|$ is the number of words in the lexicon. For all but tiny lexicons, this is simply not manageable. In order to accelerate the process, we used the *tree-count* strategy described in [8].

Unfortunately, computing $\mathcal{A}(\mathcal{L})$ for Morpho Challenge's largest lexicons still remains too time consuming.[1] Instead, we ran the analogical device on multiple languages for a week's time on randomly selected words. This enabled us to acquire a large set of analogies per language. From 11 (Arabic) to 52 (Turkish) million analogies were identified this way. While these figures may seem large at first, it is important to note that they represent but a fraction of the total potential analogies.

These sets of formal analogies are used by two systems we specifically designed for the first task of Morpho Challenge 2009. *Rali-Ana* is a pure analogical system, while *Rali-Cof* computes a set of `c-rules` (a notion we will describe shortly) which is used to accomplish the morphological analysis. The following sections describe both systems in detail.

3.1 Rali-Ana

This system makes direct use of the analogies we collected. Each time a word is involved in an analogy, we compute its factorization, as explained in Sect. 2. It is therefore possible to maintain a distribution over the *segmentations* computed for this word. The most frequent segmentation observed is kept by the system. Figure 2 illustrates the six segmentations observed for the 21 analogies involving the English word *abolishing* from which *Rali-Ana* selects *abolish+ing*.

It is important to note that because we computed only a small portion of all analogies, there are many words that this system cannot process adequately. In

[1] We roughly estimated that a few months of computation would be required for a single desk-computer to acquire all the possible analogies involving words in the Finnish lexicon for Morpho Challenge 2009.

particular, words for which no analogy is identified are added without modification to the final solution, clearly impacting recall.

abolish ing 12	ab olishing 4	abol ishing 2
a bo lishing 1	abolis hing 1	abolish in g 1

Fig. 2. Factorizations induced by analogy for the word *abolishing*. Numbers indicate the frequency of a given factorization.

3.2 Rali-Cof System

One drawback of *Rali-Ana* is that formal analogies capture information which is latent and highly lexical. For instance, knowing that [*cordial* : *cordially* = *appreciative* : *appreciatively*] does not tell us anything about [*cordial* : *appreciative* = *cordialness* : *appreciativeness*] or [*live* : *lively* = *massive* : *massively*]. Therefore, we introduce the notion of `c-rule` as a way to generalize the information captured by an analogy. Those `c-rules` are used by *Rali-Cof* in order to cluster together morphologically related words, thanks to a graph-based algorithm described hereafter.

CoFactor and C-Rule In [8], the authors introduce the notion of *cofactor* of a formal analogy $[x : y = z : t]$ as a vector of d alternations $[\langle f, g\rangle_i]_{i\in[1,d]}$ where d is the degree (see Definition 1) of the analogy and an alternation is defined formally as:

$$\langle f, g\rangle_i = \begin{cases} (f_x^{(i)}, f_z^{(i)}) \text{ if } f_x^{(i)} \equiv f_y^{(i)} \\ (f_y^{(i)}, f_z^{(i)}) \text{ otherwise} \end{cases}$$

For instance, the cofactors for our running example are: [(*cordial*,*appreciative*), (ϵ,*ly*)]. Note that the pairs of factors in this definition are not directed, that is, (ϵ, *ly*) equals (*ly*, ϵ). Cofactors such as (ϵ, *ly*) or (*ity*, *ive*) represent suffixation operations frequently involved in English. Similarly, a cofactor such as (*un*, ϵ) which might capture a prefixation operation in English (*e.g.* *loved*/*unloved*) can relate a form such as *aunt* to the form *at*, just because the former happens to contain the substring *un*. Clearly, the generalization offered by a cofactor might introduce some noise if applied blindly.

This is the motivation behind the `c-rule`, a concept we introduce in this work. A `c-rule` is a directed cofactor which is expressed as a rewriting rule $\langle \alpha \rightarrow \beta \rangle$, where α and β are the two factors of a cofactor, such that $|\alpha| \geq |\beta|$.[2] As a result, applying a `c-rule` to a word always produces a shorter one.

In order to distinguish prefixation and suffixation operations which are very frequent, we add the symbol $\star$ to the left and/or to the right of the factors in

[2] In case both factors have the same length, alphabetical ordering is used.

order to indicate the existence of a non empty factors. In our running example, the two `c-rules` $\langle \star ly \rightarrow \star\epsilon \rangle$ and $\langle appreciative\star \rightarrow cordial\star \rangle$ are collected.

For this paper, we note $\mathcal{R}(x)$, the application of the `c-rule` $\mathcal{R}$ on a word x. For instance, if $\mathcal{R}$ is $\langle \star ly \rightarrow \star\epsilon \rangle$, $\mathcal{R}(elderly)$ equals $elder$. By direct extension, we also note $[\mathcal{R}_1, \ldots, \mathcal{R}_n](x)$ the form[3] resulting from the application of n `c-rules`: $\mathcal{R}_n(\ldots \mathcal{R}_2(\mathcal{R}_1(x))\ldots)$.

Extraction of C-Rules From the set of computed analogies, we extract every `c-rule` and is frequency of occurrence. As previously stated, the number of analogies generated is huge and so is the number of `c-rules`. Therefore, we applied a filter which removes low-frequency ones.[4] Relying on counts favors `c-rules` which contain short factors. For instance in English, the `c-rule` $\langle anti\text{-}\star \rightarrow \epsilon\star \rangle$ is seen 2 472 times, while $\langle ka\star \rightarrow \epsilon\star \rangle$, which is likely fortuitous, is seen 13 839 times. To overcome this, we further score a `c-rule` $\mathcal{R}$ by its *productivity* $prod(\mathcal{R})$ defined as the ratio of the number of times its application leads to a valid form over the number of times it can be applied. Formally:

$$prod(\mathcal{R}) = |\{x \in \mathcal{L} : \mathcal{R}(x) \in \mathcal{L}\}| \ / \ |\{x \in \mathcal{L} : \mathcal{R}(x) \neq x\}|$$

Using productivity, the `c-rule` $\langle anti\text{-}\star \rightarrow \epsilon\star \rangle$ has a score of 0.9490 compared to 0.2472 for $\langle ka\star \rightarrow \epsilon\star \rangle$.

Word Relation Trees (WRT) construction *Rali-Cof* builds a forest of WRTs, where each tree identifies morphologically related words. A WRT is a structure where the nodes are the lexicon's words. An edge between nodes n_a and n_b, noted $n_a \rightarrow n_b$, is labelled by a set of `c-rules` which transforms word n_a into word n_b. The construction of the WRT forest is a greedy process, which applies the three following steps until all words in the lexicon have been processed:

1. Pick untreated word n from the lexicon.[5]
2. Compute set $\mathcal{S}(n)$ which contains words that can be reached by applying any strictly positive number of `c-rules` to word n.
3. Add an edge from n to $b \equiv \text{argmax}_{w \in \mathcal{S}(n)}\, score(n, w)$, the word of $\mathcal{S}(n)$ which maximizes a score (described hereafter), provided this score is greater than a given threshold.[6]

While building $\mathcal{S}(n)$ during step 2, it is often the case that different paths from word n to word w exist, as illustrated in Fig. 3. Therefore, the score between two words is computed by summing the score of each path. In turn, the score of one path $[\mathcal{R}_1, \ldots, \mathcal{R}_m]$, where $[\mathcal{R}_1, \ldots, \mathcal{R}_m](n) \equiv w$, is computed as $\prod_{i=1}^{m} prod(\mathcal{R}_i)$. If w happens to be the word selected at step 3, the retained path becomes an edge in the WRT labelled by the sequence of `c-rules` leading word n to word w.

[3] For the sake of clarity, we omit the case where the application of a `c-rule` leads to several forms.

[4] C-rules occurring less than 20 times are removed.

[5] The order in which the words are considered is unimportant.

[6] Set to 0.35 in this study.

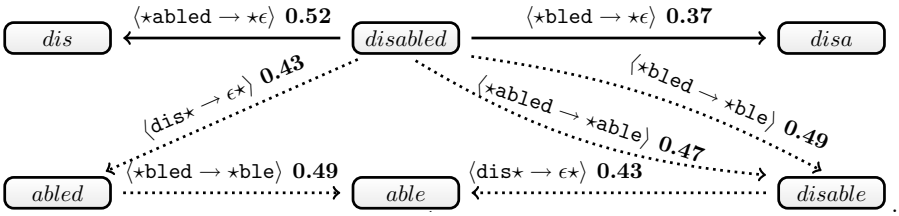

Fig. 3. Graph for the word `disabled`. The most probable link is `disable` with a score of 0.96. The dotted edges indicate the path considered for the computation of the score between `disabled` and `able`.

Segmentation into Morphemes Each node in a WRT contains the segmentation of its associated word into its morphemes. In case of the root node, the set of morphemes is a singleton containing the word itself. For any other node (n), the set of morphemes is obtained by grouping together the morphemes of the father node (f) and those involved in the `c-rules` labeling the edge $n \rightarrow f$. To take one simple example, imagine a WRT contains the edge `disabled` $\rightarrow$ `able`, labelled by [$\langle\epsilon \rightarrow$ `dis`$\star\rangle$,$\langle\epsilon \rightarrow \star$ `d`$\rangle$]. The morphemes of `disabled` are [`able`,`dis`,`d`], where `dis` and `d` are the two morphemes present in the `c-rules`. As intuitive as it seems, the segmentation process involves intricate measures, the details of which are omitted for the sake of simplicity.

4 Experiments

The evaluation of the two systems we designed has been conducted by the Morpho Challenge 2009 organizers. The details of the evaluation protocol and the results can be found in [6]. Table 1 gives the official performance of our two systems compared to the one of *Morfessor* [3], a widely used system also employed as a baseline in Morpho Challenge. As can be observed, *Rali-Cof* outperforms both *Rali-Ana* and *Morfessor* for Finnish, Turkish and German.

The low recall of *Rali-Ana* can be explained by the fact that only a small subset of the analogies have been identified (See Sect.3.1). Nevertheless, the results yielded by this system are encouraging considering its simplicity. Especially since the precision for each language is rather good. We know that if we compute more analogies, recall will increase with the lexicon's coverage. Since the analyzed words were chosen without bias, precision will predictably not change much.

We observe that *Rali-Cof*'s performances are similar for all languages except for Arabic, for which we have a low recall. This might be caused by the provided lexicon's size, which is over 10 times inferior to that of the next smallest. Since analogical learning relies on the pattern frequency to identify morphemes, several valid morphemes might be overlooked due to their low frequency in the training set.

Although *Morfessor* has a higher F-Score in English, our approach surpasses it for languages with higher morphological complexity. This is noteworthy as the potential benefit of morphological analysis is greater for those languages.

5 Discussion and future work

We have presented the two systems we designed for our participation in Morpho Challenge 2009. While both use formal analogy, *Rali-Cof* extracts the lexicalized information captured by an analogy through the use of `c-rules` a concept we introduced here. While *Rali-Ana* requires computing the full set of analogies involving the words found in a lexicon, *Rali-Cof* only requires a (small) subset of those analogies to function correctly and is therefore more practical.

Considering only a fraction of the total available words have been processed by *Rali-Ana*, its performances are rather promising. We are also pleased to note that *Rali-Cof* outperforms a fair baseline(*Morfessor*) on Turkish, Finnish and German.

We developped our systems within a very short period of time, making many hard decisions that we did not have time to investigate further. This reinforces our belief that formal analogies represent a principled concept that can be efficiently used for unsupervised morphology acquisition.

Still, a number of avenues remain to be investigated. First, we did not adjust the meta-parameters controlling the *Rali-Cof* system to a specific language. This could be done using a small supervision set, that is, a set of words that are known to be morphologically related. Second, we plan to investigate the impact of the quantity of analogies computed. Preliminary experiments showed that in English, formal analogies computed on less than 10% of the words in the lexicon could identify most of the major affixes. Third, while `c-rules` capture more context than cofactors do, other alternatives might be considered, such as regular expressions, as in [16]. Last, we observed that sometimes, words in a WRT are not morphologically related. We think it is possible to consider formal analogies in order to filter out some associations made while constructing the WRT forest.

Table 1. Precision (Pr.), Recall (Rc.) and F-measure (F1) for our systems and for the reference system, *Morfessor*, in the Morpho Challenge 2009 workshop.

	Rali-Cof			*Rali-Ana*			*Morfessor Baseline*		
	Pr.	Rc.	F1	Pr.	Rc.	F1	Pr.	Rc.	F1
ENG.	68.32	46.45	55.30	64.61	33.48	44.10	**74.93**	**49.81**	**59.84**
FIN.	**74.76**	**26.20**	**38.81**	60.06	10.33	17.63	89.41	15.73	26.75
TUR.	**48.43**	**44.54**	**46.40**	69.52	12.85	21.69	89.68	17.78	29.67
GER.	**67.53**	**34.38**	**45.57**	61.39	15.34	24.55	81.70	22.98	35.87
ARB.	94.56	2.13	4.18	92.40	4.40	8.41	**91.77**	**6.44**	**12.03**

References

1. Harris, Z.S.: From phoneme to morpheme. Language **31**(2) (1955) 190–222
2. Bernhard, D.: Simple morpheme labelling in unsupervised morpheme analysis. In: CLEF 2007 Workshop, Budapest, Hungary (Sept. 2007) 873–880
3. Creutz, M., Lagus, K.: Inducing the morphological lexicon of a natural language from unannotated text. In: In Proceedings of AKRR'05. Volume 5. (2005) 106–113
4. Monson, C., Carbonell, J., Lavie, A., Levin, L.: Paramor: Minimally supervised induction of paradigm structure and morphological analysis. In: Proceedings of 9th SIGMORPHON Workshop, Prague, Czech Republic, ACL (June 2007) 117–125
5. Zeman, D.: Using unsupervised paradigm acquisition for prefixes. In: CLEF 2008 Workshop, Arhus, Denmark (Sept. 2008)
6. Kurimo, M., Virpioja, S., Turunen, V., Blackwood, G., Byrne, W.: Overview and results of morpho challenge 2009. In: 10th CLEF Workshop, Corfu,Greece (2010)
7. Lepage, Y., Denoual, E.: Purest ever example-based machine translation: Detailed presentation and assessment. Machine Translation **29** (2005) 251–282
8. Langlais, P., Patry, A.: Translating unknown words by analogical learning. In: EMNLP-CoNLL, Prague, Czech Republic (June 2007) 877–886
9. Denoual, E.: Analogical translation of unknown words in a statistical machine translation framework. In: MT Summit, XI, Copenhagen (Sept. 10-14 2007)
10. Stroppa, N., Yvon, F.: An analogical learner for morphological analysis. In: CoNLL, Ann Arbor, MI (June 2005) 120–127
11. Hathout, N.: From wordnet to celex: acquiring morphological links from dictionaries of synonyms. In: 3rd LREC, Las Palmas de Gran Canaria (2002) 1478–1484
12. Hathout, N.: Acquistion of the morphological structure of the lexicon based on lexical similarity and formal analogy. In: 3rd Textgraphs workshop, Manchester, United Kingdom (Aug. 2008) 1–8
13. Pirrelli, V., Yvon, F.: The hidden dimension: a paradigmatic view of data-driven NLP. Journal of Experimental & Theroretical Artifical Intelligence **11** (1999) 391–408
14. Yvon, F., Stroppa, N., Delhay, A., Miclet, L.: Solving analogical equations on words. Technical Report D005, ENST, Paris, France (Jul. 2004)
15. Stroppa, N.: Définitions et caractérisations de modèles à base d'analogies pour l'apprentissage automatique des langues naturelles. PhD thesis, ENST, ParisTech, Télécom, Paris, France (Nov. 2005)
16. Bernhard, D.: Morphonet: Exploring the use of community structure for unsupervised morpheme analysis. In: 10th CLEF Workshop, Corfu,Greece (2010)