Mapping Source to Target Strings without Alignment by Analogical Learning: A Case Study with Transliteration

Philippe Langlais
RALI / DIRO
Université de Montréal
Montréal, Canada, H3C 3J7
felipe@iro.umontreal.ca

Abstract

Analogical learning over strings is a holistic model that has been investigated by a few authors as a means to map forms of a source language to forms of a target language. In this study, we revisit this learning paradigm and apply it to the transliteration task. We show that alone, it performs worse than a statistical phrase-based machine translation engine, but the combination of both approaches outperforms each one taken separately, demonstrating the usefulness of the information captured by a so-called formal analogy.

1 Introduction

A proportional analogy is a relationship between four objects, noted $[x : y :: z : t]$, which reads as "x is to y as z is to t". While some strategies have been proposed for handling semantic relationships (Turney and Littman, 2005; Duc et al., 2011), we focus in this study on formal proportional analogies (hereafter formal analogies or simply analogies), that is, proportional analogies involving relationships at the graphemic level, such as [*atomkraftwerken* : *atomkriegen* :: *kraftwerks* : *kriegs*] in German.

Analogical learning over strings has been investigated by several authors. Yvon (1997) addressed the task of grapheme-to-phoneme conversion, a problem which continues to be studied actively, see for instance (Bhargava and Kondrak, 2011). Stroppa and Yvon (2005) applied analogical learning to computing morphosyntactic features to be associated with a form (lemma, part-of-speech, and additional features such as number, gender, case, tense, mood, etc.). The performance of the analogical engine on the Dutch language was as good as or better than the one reported in (van den Bosch and Daelemans, 1993). Lepage and Denoual (2005) pioneered the application of analogical learning to Machine Translation. Different variants of the system they proposed have been tested in a number of evaluation campaigns, see for instance (Lepage et al., 2009). Langlais and Patry (2007) investigated the more specific task of translating unknown words, a problem simultaneously studied in (Denoual, 2007).

Analogical learning has been applied to various other purposes, among which query expansion in information retrieval (Moreau et al., 2007), classification of nominal and binary data, and handwritten character recognition (Miclet et al., 2008). Formal analogy has also been used for solving Raven IQ tests (Correa et al., 2012).

In this study, we investigate the relevance of analogical learning for English proper name transliteration into Chinese. We compare it to the statistical phrase-based machine translation approach (Koehn et al., 2003) initially proposed for transliteration by Finch and Sumita (2010). We show that alone, analogical learning underperforms the phrase-based approach, but that a combination of both outperforms individual systems.

We describe in section 2 the principle of analogical learning. In section 3, we report on experiments we conducted in applying analogical learning on the NEWS 2009 English-to-Chinese transliteration task. Related works are discussed in section 4. We conclude in section 5 and identify avenues we believe deserve investigations.

2 Analogical Learning

2.1 Formal Analogy

In this study, we use the most general definition of formal analogy we found, initially described in (Yvon et al., 2004). It handles a large variety of relations, including but not limited to affixation operations (i.e. [*capital* : *anticapitalisme* :: *commun* : *anticommuniste*] in French), stem mu-

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 684–689,
Sofia, Bulgaria, August 4-9 2013.

tations (i.e. [*lang* : *länge* :: *stark* : *stärke*] in German), and even templatic relations (i.e [*KaaTiB* : *KuTaaB* :: *QaaRi'* : *QuRaa'*] in Arabic).

Informally,[1] this definition states that 4 forms x, y, z and t are in analogical relation iff we can find a d-factorization (a factorization into d factors) of each form, such that the i^{th} factors ($i \in [1, d]$) of x and z equal (in ensemble terms) the i^{th} factors of y and t.

For instance, [*this guy drinks* : *this boat sinks* :: *these guys drank* : *these boats sank*] holds because of the following 4-uple of 5-factorizations, whose factors are aligned column-wise for clarity, and where spaces (underlined) are treated as regular characters (ϵ designates the empty factor):

f_x	$\equiv$	(	*this*	*_guy*	ϵ	*_dr*	*inks*	)
f_y	$\equiv$	(	*this*	*_boat_*	ϵ	*s*	*inks*	)
f_z	$\equiv$	(	*these*	*_guy*	*s*	*_dr*	*ank*	)
f_t	$\equiv$	(	*these*	*_boat_*	*s*	*s*	*ank*	)

This analogy "captures" among other things that in English, changing *this* for *these* implies a plural mark (*s*) to the corresponding noun. Note that analogies can relate arbitrarily distant substrings. For instance the 3rd-person singular mark of the verbs relates to the first substring *this*.

2.2 Analogical Learning

We now clarify the process of analogical learning. Let $\mathcal{L} = \{(i(x_k), o(x_k))_k\}$ be a training set (or memory) gathering pairs of input $i(x_k)$ and output $o(x_k)$ representations of elements x_k. In this study, the elements we consider are pairs of English / Chinese proper names in a transliteration relation. Given an element t for which we only know $i(t)$, analogical learning works by:

1. building $\mathcal{E}_i(t) = \{(x, y, z) \in \mathcal{L}^3 \mid [i(x) : i(y) :: i(z) : i(t)]\}$, the set of triples in the training set that stand in analogical proportion with t in the input space,
2. building $\mathcal{E}_o(t) = \{[o(x) : o(y) :: o(z) : ?] \mid (x, y, z) \in \mathcal{E}_i(t)\}$, the set of solutions to the output analogical equations obtained,
3. selecting $o(t)$ among the solutions aggregated into $\mathcal{E}_o(t)$.

In this description, we define an *analogical equation* as an analogy with one form missing, and we note [$x : y :: z : ?$] the set of its solutions (i.e. *undoable* $\in$ [*reader* : *doer* :: *unreadable* : *?*]).[2]

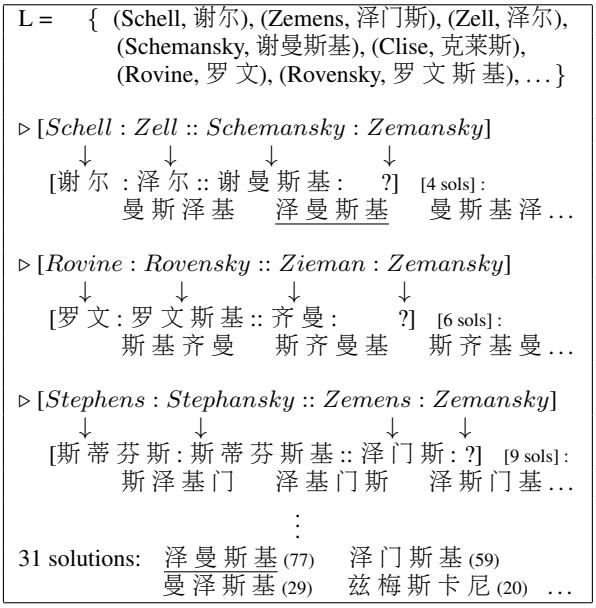

Figure 1: Excerpt of a transliteration session for the English proper name *Zemansky*. 31 solutions have been identified in total (4 by the first equation reported); the one underlined (actually the most frequently generated) is the sanctioned one.

Figure 1 illustrates this process on a transliteration session for the English proper name *Zemansky*. The training corpus $\mathcal{L}$ is a set of pairs of English proper names and their Chinese Transliteration(s). Step 1 identifies analogies among English proper names: 7 such analogies are identified, 3 of which are reported (marked with a ▷ sign). Step 2 projects the English forms in analogical proportion into their known transliteration (illustrated by a ↓ sign) in order to solve Chinese analogical equations. Step 3 aggregates the solutions produced during the second step. In the example, it consists in sorting the solutions in decreasing order of the number of time they have been generated during step 2 (see next section for a better strategy).

There are several important points to consider when deploying the learning procedure shown above. First, the search stage (step 1) has a time complexity that can be prohibitive in some applications of interest. We refer the reader to (Langlais and Yvon, 2008) for a practical solution to this. Second, we need a way to solve an analogical

[1] We refer the reader to (Stroppa and Yvon, 2005) for a more technical exposition.

[2] Analogical equation solvers typically produce several solutions to an equation.

equation. We applied the finite-state machine procedure described in (Yvon et al., 2004). Suffice it to say that typically, this solver produces several solutions to an equation, most of them spurious,[3] reinforcing the need for an efficient aggregation step (step 3). Last, it might happen that the overall approach fails at producing a solution, because no input analogy is identified during step 1, or because the input analogies identified do not lead to analogies in the output space. This *silence* issue is analyzed in section 3. A detailed account of those problems and possible solutions are discussed in (Somers et al., 2009).

We underline that analogies in both source and target languages are considered *independently*: the approach does not attempt to align source and target substrings, but relies instead on the inductive bias that input analogies (often) imply output ones.

3 Experiments

3.1 Setting

The task we study is part of the NEWS evaluation campaign conducted in 2009 (Li et al., 2009). The dataset consists of 31 961 English-Chinese transliteration examples for training the system (TRAIN), 2 896 ones for tuning it (DEV), and 2 896 for testing them (TEST).

We compare two different approaches to transliteration: a statistical phrase-based machine translation engine — which according to Li et al. (2009) was popular among participating systems to NEWS — as well as differently flavored analogical systems.

We trained (on TRAIN) a phrase-based translation device with the Moses toolkit (Koehn et al., 2007), very similarly to (Finch and Sumita, 2010), that is, considering each character as a word. The coefficients of the log-linear function optimized by Moses' decoder were tuned (with MERT) on DEV.

For the analogical system, we investigated the use of classifiers trained in a supervised way to recognize the good solutions generated during step 2. For this, we first transliterated the DEV dataset using TRAIN as a memory. Then, we trained a classifier, taking advantage of the DEV corpus for the supervision. We tried two types of learners — support vector machines (Cortes and Vapnik, 1995) and voted perceptrons (Freund and Schapire, 1999)[4] — and found the former to slightly outperform the latter. Finally, we transliterated the TEST corpus using both the TRAIN and DEV corpora as a memory,[5] and applied our classifiers on the solutions generated.

The lack of space prevents us to describe the 61 features we used for characterizing a solution. We initially considered a set of features which characterizes a solution (frequency, rank in the candidate list, language model likelihood, etc.), and the process that generated the solution (i.e. number of analogies involved), but no feature that would use scored pairs of substrings (such as mutual information of substrings).[6] Thus, we also considered in a second stage a set of features that we collected thanks to a n-best list of solutions computed by Moses (Moses' score given to a solution, its rank in the n-best list, etc.).

3.2 Results

We ran the NEWS 2009 official evaluation script[7] in order to compute ACC (the accuracy of the first solution), F_1 (the F-measure which gives partial credits proportional to the longest subsequence between the reference transliteration and the first candidate), and the Mean Reciprocal Rank (MRR), where 100/MRR roughly indicates the average rank of the correct solution over the session.

Table 1 reports the results of several transliteration configurations we tested. The first two systems are pure analogical devices, (M) is the Moses configuration, (AM_1) is a variant discussed further, (AM_2) is the best configuration we tested (a combination of Moses and analogical learning), and the last two lines show the lowest and highest performing systems among the 18 standard runs registered at NEWS 2009 (Li et al., 2009). Several observations have to be made.

First, none of the variants tested outperformed the best system reported at NEWS 2009. This is not surprising since we conducted only preliminary experiments with analogy. Still, we were pleased to observe that the best configuration we devised (AM_2) would have ranked fourth on this task.

[3] A spurious solution is a string that does not belong to the language under consideration. See Figure 1 for examples.

[4] We used libSVM (Chang and Lin, 2011) for training svms, and an in-house package for training voted perceptrons.

[5] This is fair since there is no training involved. Many participants to the NEWS campaign did this as well.

[6] We avoided this in order to keep the classifiers simple to train.

[7] http://translit.i2r.a-star.edu.sg/news2009/evaluation/.

The ana-freq system is an analogical device where the aggregation step consists in sorting solutions in decreasing order of frequency. It is clearly outperformed by the Moses system. The ana-svm$_a$ system is an analogical device where the solutions are selected by the SVM trained on analogical features only. Learning to recognize good solutions from spurious ones improves accuracy (over A$_1$). Still, we are far from the accuracy we would observe by using an oracle classifier (ACC = 81.5). Clearly, further experiments with better feature engineering must be conducted. It is noteworthy that the pure analogical devices we tested (A$_1$ and A$_2$) did not return any solution for 3.7% of the test forms, which explains some loss in performance compared to the SMT approach, which always delivers a solution.[8]

System ana-svm$_{a+m}$ (AM$_1$) is an analogical device where the classifier makes uses of the features extracted by Moses. Obviously, those features drastically improve accuracy of the classifier. Configuration (AM$_2$) is a combination which cascades the hybrid device (AM$_1$) with the SMT engine (M). This means that the former system is trusted whenever it produces a solution, and the latter one is used as a backup. This configuration outperforms Moses, which demonstrates the complementarity of the analogical information.

Configuration		ACC	F_1	MRR	*rank*
A$_1$	ana-freq	56.6	79.1	63.0	16
A$_2$	ana-svm$_a$	58.0	80.0	58.8	15
M	moses	66.6	85.9	66.6	6
AM$_1$	ana-svm$_{a+m}$	63.4	82.0	64.1	10
AM$_2$	AM$_1$ + M	68.5	86.9	69.0	4
	last NEWS 2009	19.9	60.6	22.9	23
	first NEWS 2009	73.1	89.5	81.2	1

Table 1: Evaluation of different configurations with the metrics used at NEWS. The last column indicates the rank of systems as if we had submitted the top 5 configurations to NEWS 2009.

4 Related Work

Most approaches to transliteration we know rely on some form of substring alignment. This alignment can be learnt explicitly as in (Knight and Graehl, 1998; Li et al., 2004; Jiampojamarn et al., 2007), or it can be indirectly modeled as in (Oh et al., 2009) where transliteration is seen as a tagging task (that is, labeling each source grapheme with a target one), and where the model learns correspondences at the substring level. See also the semi-supervised approach of (Sajjad et al., 2012). Analogical inference differs drastically from those approaches, since it finds relations in the source material and solves target equations independently. Therefore, no alignment whatsoever is required.

Transliteration by analogical learning has been attempted by Dandapat et al. (2010) for an English-to-Hindi transliteration task. They compared various heuristics to speed up analogical learning, and several combinations of phrase-based SMT and analogical learning. Our results confirm the observation they made that combining an analogical device with SMT leads to gains over individual systems. Still, their work differs from the present one in the fact that they considered the top frequency aggregator (similar to A$_1$), which we showed to be suboptimal. Also, they used the definition of formal analogy of Lepage (1998), which is provably less general than the one we used. The impact of this choice for different language pairs remains to be investigated.

Aggregating solutions produced by analogical inference with the help of a classifier has been reported in (Langlais et al., 2009). The authors investigated an arguably more specific task: translating medical terms. Another difference is that we classify *solutions* produced by analogical learning (roughly 100 solutions per test form), while they classified *pairs of input/target analogies*, whose number can be rather high, leading to huge and highly unbalanced learning tasks. The authors report training experiments with millions of examples and only a few positive ones. In fact, we initially attempted to recognize fruitful analogical pairs, but found it especially slow and disappointing.

5 Conclusion

We considered the NEWS 2009 English-to-Chinese transliteration task for investigating analogical learning, a holistic approach that does not rely on an alignment or segmentation model. We have shown that alone, the approach fails to translate 3.7% of the test forms, underperforms the state-of-the-art SMT engine Moses, while still de-

[8] Removing the solutions produced by the SMT engine for the 3.7% test forms that receive no solution from the analogical devices would result in an accuracy score of 65.0.

livering decent performance. By combining both approaches, we obtained a system which outperforms the individual ones we tested.

We believe analogical inference over strings has not delivered all his potential yet. In particular, we have observed that there is a huge room for improvements in the aggregation step. We have tested a simple classifier approach, mining a tiny subset of the features that could be put at use. More research on this issue is warranted, notably looking at machine-learned ranking algorithms.

Also, the silence issue we faced could be tackled by the notion of *analogical dissimilarity* introduced by Miclet et al. (2008). The idea of using near analogies in analogical learning has been successfully investigated by the authors on a number of standard classification testbeds.

Acknowledgments

This work has been founded by the Natural Sciences and Engineering Research Council of Canada. We are grateful to Fabrizio Gotti for his contribution to this work, and to the anonymous reviewers for their useful comments. We are also indebted to Min Zhang and Haizhou Li who provided us with the NEWS 2009 English-Chinese datasets.

References

Aditya Bhargava and Grzegorz Kondrak. 2011. How do you pronounce your name? Improving G2P with transliterations. In *49th ACL/HLT*, pages 399–408, Portland, USA.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):1–27, May.

William Fernando Correa, Henri Prade, and Gilles Richard. 2012. When intelligence is just a matter of copying. In *20th ECAI*, pages 276–281, Montpellier, France.

Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. *Mach. Learn.*, 20(3):273–297.

Sandipan Dandapat, Sara Morrissey, Sudip Kumar Naskar, and Harold Somers. 2010. Mitigating Problems in Analogy-based EBMT with SMT and vice versa: a Case Study with Named Entity Transliteration. In *24th Pacific Asia Conference on Language Information and Computation (PACLIC'10)*, pages 365–372, Sendai, Japan.

Étienne Denoual. 2007. Analogical translation of unknown words in a statistical machine translation framework. In *MT Summit XI*, pages 135–141, Copenhagen, Denmark.

Nguyen Tuan Duc, Danushka Bollegala, and Mitsuru Ishizuka. 2011. Cross-Language Latent Relational Search: Mapping Knowledge across Languages. In *25th AAAI*, pages 1237 – 1242, San Francisco, USA.

Andrew Finch and Eiichiro Sumita. 2010. Transliteration Using a Phrase-based Statistical Machine Translation System to Re-score the Output of a Joint Multigram Model. In *2nd Named Entities Workshop (NEWS'10)*, pages 48–52, Uppsala, Sweden.

Y. Freund and R. E. Schapire. 1999. Large Margin Classification Using the Perceptron Algorithm. *Mach. Learn.*, 37(3):277–296.

Sittichai Jiampojamarn, Grzegorz Kondrak, and Tarek Sherif. 2007. Applying Many-to-Many Alignments and Hidden Markov Models to Letter-to-Phoneme Conversion. In *NAACL/HLT'07*, pages 372–379.

Kevin Knight and Jonathan Graehl. 1998. Machine Transliteration. *Comput. Linguist.*, 24(4):599–612.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *NAACL/HLT'03*, pages 48–54, Edmonton, Canada.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *45th ACL*, pages 177–180. Interactive Poster and Demonstration Sessions.

Philippe Langlais and Alexandre Patry. 2007. Translating Unknown Words by Analogical Learning. In *EMNLP/CoNLL'07*, pages 877–886, Prague, Czech Republic.

Philippe Langlais and François Yvon. 2008. Scaling up Analogical Learning. In *22nd COLING*, pages 51–54, Manchester, United Kingdom. Poster.

Philippe Langlais, François Yvon, and Pierre Zweigenbaum. 2009. Improvements in Analogical Learning: Application to Translating multi-Terms of the Medical Domain. In *12th EACL*, pages 487–495, Athens, Greece.

Yves Lepage and Étienne Denoual. 2005. Purest ever example-based machine translation: Detailed presentation and assesment. *Mach. Translat*, 19:25–252.

Yves Lepage, Adrien Lardilleux, and Julien Gosme. 2009. The GREYC Translation Memory for the IWSLT 2009 Evaluation Campaign: one step beyond translation memory. In *6th IWSLT*, pages 45–49, Tokyo, Japan.

Yves Lepage. 1998. Solving Analogies on Words: an Algorithm. In *COLING/ACL*, pages 728–733, Montreal, Canada.

Haizhou Li, Min Zhang, and Jian Su. 2004. A Joint Source-Channel Model for Machine Transliteration. In *42nd ACL*, pages 159–166, Barcelona, Spain.

Haizhou Li, A. Kumaran, Vladimir Pervouchine, and Min Zhang. 2009. Report of NEWS 2009 Machine Transliteration Shared Task. In *1st Named Entities Workshop (NEWS'09): Shared Task on Transliteration*, pages 1–18, Singapore.

Laurent Miclet, Sabri Bayroudh, and Arnaud Delhay. 2008. Analogical Dissimilarity: Definitions, Algorithms and two experiments in Machine Learning. *Journal of Artificial Intelligence Research*, pages 793–824.

Fabienne Moreau, Vincent Claveau, and Pascale Sébillot. 2007. Automatic Morphological Query Expansion Using Analogy-based Machine Learning. In *29th European Conference on IR research (ECIR'07)*, pages 222–233, Rome, Italy.

Jong-hoon Oh, Kiyotaka Uchimoto, and Kentaro Torisawa. 2009. Machine Transliteration using Target-Language Grapheme and Phoneme: Multi-engine Transliteration Approach. In *1st Named Entities Workshop (NEWS'09): Shared Task on Transliteration*, pages 36–39, Singapore.

Hassan Sajjad, Alexander Fraser, and Helmut Schmid. 2012. A Statistical Model for Unsupervised and Semi-supervised Transliteration Mining. In *50th ACL*, pages 469–477, Jeju Island, Korea.

Harold Somers, Sandipan Sandapat, and Sudip Kumar Naskar. 2009. A Review of EBMT Using Proportional Analogies. In *3rd Workshop on Example-based Machine Translation*, pages 53–60, Dublin, Ireland.

Nicolas Stroppa and François Yvon. 2005. An Analogical Learner for Morphological Analysis. In *9th CoNLL*, pages 120–127, Ann Arbor, USA.

P.D. Turney and M.L. Littman. 2005. Corpus-based Learning of Analogies and Semantic Relations. In *Machine Learning*, volume 60, pages 251–278.

Antal van den Bosch and Walter Daelemans. 1993. Data-Oriented Methods for Grapheme-to-Phoneme Conversion. In *6th EACL*, pages 45–53, Utrecht, Netherlands.

François Yvon, Nicolas Stroppa, Arnaud Delhay, and Laurent Miclet. 2004. Solving Analogies on Words. Technical Report D005, École Nationale Supérieure des Télécommuncations, Paris, France.

François Yvon. 1997. Paradigmatic Cascades: a Linguistically Sound Model of Pronunciation by Analogy. In *35th ACL*, pages 429–435, Madrid, Spain.