Extraction of analogies between sentences on the level of syntax using parse trees

Yifei Zhou[1,*], Rashel Fam[1] and Yves Lepage[1]

[1]Waseda University, 2-7 Hibikino, Kitakyushu, 808-0135, Japan

Abstract

Example-based machine translation by analogy is an alternative approach to machine translation. Its principle is relatively simple, but the absolute number of analogies between sentences contained in the corpus is crucial for the overall quality of translation. The relative number of analogies is called the analogical density. The goal of this paper is to measure the analogical density of different aligned corpora. To this end, we extract analogies between sentences. Now, we use parse trees to represent sentences on the level of syntax. We report analogical densities for five different languages in an aligned multilingual corpus extracted from the Tatoeba resource, at the level of characters, words or parse trees.

Keywords

Sentence analogy, parse tree, example-based machine translation

1. Introduction

Analogy is known to be an essential skill in human cognition. It can be used to interpret or analyze words or sentences that are unfamiliar or have never been seen before [1, 2, 3, 4, 5]. In other words, analogy has the power to explain the unknown using the known. Analogy can play a role in natural language processing tasks such as machine translation [6, 7, 8], transliteration [9, 10] or question answering [11].

Example-Based Machine Translation (EBMT) by analogy implements a case-based reasoning approach to machine translation [12]. It generates translations relying on analogies in the source language and the target language after retrieval of similar sentences from a knowledge database. There, analogy exploits examples (cases) contained in the knowledge container (case base) to solve unknown cases.

By denoting $A : B :: C : D$ the analogical relationship between four sentences: A, B, C and D, Formula (1) defines sentence analogies in two languages with sentences which are translations of one another. $A : B :: C : D$ denotes a monolingual analogy in the source language and $A' : B' :: C' : D'$ is corresponding translation in the target language. Figure 1 instantiates Formula (1) on an example in English and French.

ICCBR Analogies'22: Workshop on Analogies: from Theory to Applications at ICCBR-2022, September, 2022, Nancy, France

*Corresponding author.

yifei.zhou@ruri.waseda.jp (Y. Zhou); fam.rashel@fuji.waseda.jp (R. Fam); yves.lepage@waseda.jp (Y. Lepage)

$$\begin{array}{ccccccc} A & : & B & :: & C & : & D \\ \updownarrow & & \updownarrow & & \updownarrow & & \updownarrow \\ A' & : & B' & :: & C' & : & D' \end{array} \tag{1}$$

I like apples.	:	*I don't like apples.*	::	*I speak Swedish.*	:	*I don't speak Swedish.*
↕		↕		↕		↕
J'aime les pommes.	:	*Je n'aime pas les pommes.*	::	*Je parle le suédois.*	:	*Je ne parle pas le suédois.*

Figure 1: Two corresponding monolingual analogies between sentences in English and French

The number of analogies that exist in a given corpus is crucial for EBMT by analogy. Our objective in the present work is to estimate the number of analogies similar to the one shown in Figure 1, for various language pairs. Now, analogies can be extracted at various levels: surface form or syntax. To extract analogies automatically, we use vector representations of sentences based on the occurrence of characters, tokens, or branches in parse trees. We then count the number of extracted analogies and can compute the *analogical density* of the corpus. Although we do not conduct experiments in this paper, our intuition is that a higher number of analogies will lead to better translations in an EBMT system by analogy.

2. Related Work

2.1. Traditional Levels: Formal and Semantic Analogies

Formal analogies do not take into account the meaning or the syntax of sentences. Instead, the surface form, i.e., characters or words, are only taken into account. [13] uses *abc* : *abbccd* :: *efg* : *effggh* as an example to clarify what formal analogy is. The changes are only between characters and the strings bear no meaning. *walk* : *walked* :: *go* : *goed* is another instance of formal analogy: *goed* is not a valid English word form for the simple past tense form of *go*. However, on the level of form, the analogy holds: the suffix *-ed* has just been added at the end of the string *go*, as for *walk*.

In an analogy at the semantic level, the meaning attached to the strings is considered. For instance, *king* : *queen* :: *man* : *woman* is a classic example of semantic analogy [14]. It exhibits the male / female opposition. In contrast to the previous formal analogy, *walk* is to *walked* as *go* is to *went* is valid on the level of meaning, or rather grammar. Table 1 shows examples of analogies between sentences on one of the two levels of form or meaning, or both.

2.2. Between Form and Semantics: Syntax

In the present paper, we concentrate on analogies between sentences. In [15, 16, 17], a method to perform syntactic analysis of sentences, i.e., to obtain a syntax tree for a given sentence, has

Table 1
Example of analogies between sentences on the level of form and meaning

Analogy between sentences								Level	
								Form	Meaning
They work hard.	:	*He worked very hard.*	::	*They look happy.*	:	*He looked very happy.*		yes	yes
The boy speaks Thai.	:	*The girl goes to Thailand.*	::	*The actor spoke Chinese.*	:	*The actress went to China.*		no	yes
I talk to him.	:	*I talked to him.*	::	*I go to school.*	:	*I goed to school.*		yes	no

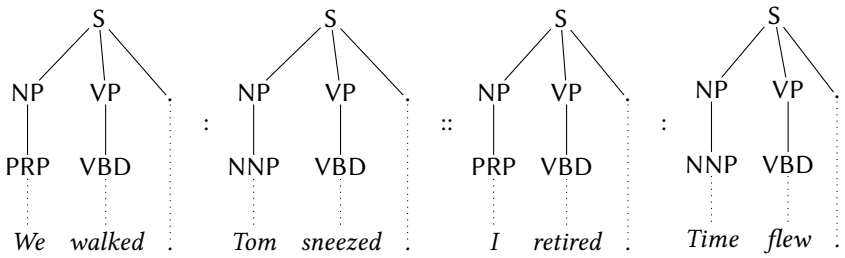

Figure 2: Analogy between sentences on the level of syntax using constituency representation

been described. It relied on the use of analogy. Similarly, an example of an analogy between syntactic trees is shown in [18, 19]. It corresponds to an active passive transformation between sentences: the analogy holds not only on the level of form, but also on the level of syntax.

[18, 19] show that syntactic representations of sentences can be used as yet another level to capture analogies between sentences, in addition to the formal and semantic levels. However, analogy on the level of syntax is different from both the formal and semantic levels. It is well known that grammaticality is independent from meaning, as illustrated by the classic example sentence: *Colorless green ideas sleep furiously* [20].

We propose to work on analogy at the level of syntax. Figure 2 is another example of a syntactic analogy between sentences. There, the sentences do not acceptedly create an analogy on the level of form or meaning, but they definitely make an analogy at the syntactic level: exchange of personal pronoun (PRP) with proper noun (NNP). Notice that, for the analogy to hold, the terminals (the words in the sentences) which should appear on the leaves in the parse trees are not considered.

3. Analogy on the Level of Syntax Using Parse Trees

While past studies concentrated on analogies on the formal level, the originality of this paper is to extract and count analogies between sentences on the level of syntax using parse trees. To this end, we develop two components. The first component computes vector representations. A sentence is represented by a feature vector counting the number of occurrences of all branches in any parse tree of a sentence from the corpus considered. The second component computes

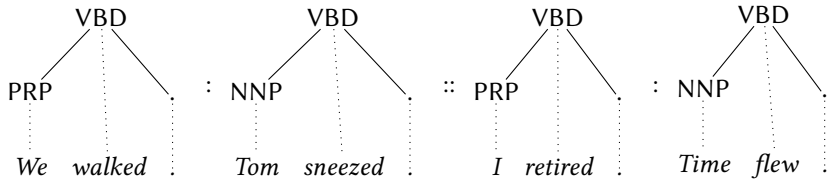

Figure 3: Analogy between sentences on the level of syntax using dependency representation

the ratio between these vectors of features on two given trees. The ratio between sentences is simply defined as the difference between their feature vectors.

3.1. Tree Representations

In computational linguistics, a parse tree is a tree that represents the syntactic structure of a sentence [21]. In constituency parse trees, the tree reflects the grouping of words in a sentence by constituents or phrases. In dependency parse trees, the branches show the dependency relationship between words. We use Universal Dependency parsers provided by the spaCy[1] library, for various languages, and converted all sentences in our corpora into dependency parse trees. The dependency parse trees of the sentences in Figure 2 are shown in Figure 3.

A sentence S can be represented by a feature vector $\overrightarrow{T_S}$ by counting the number of occurrences for all the branches found in its parse tree T_S. In Formula (2), the notation $|T_S|_{branch}$ stands for the number of times a *branch* appears in the parse tree T_S of sentence S.

$$\overrightarrow{T_A} = \begin{pmatrix} |T_A|_{VBD \to PRP} \\ |T_A|_{VBD \to NNP} \\ \vdots \\ |T_A|_{VBD \to .} \end{pmatrix} \tag{2}$$

3.2. Ratios between Trees

To extract analogies at the level of syntax, we calculate the ratio between trees. Formula (3) defines the ratio between sentences A and B as the difference between their vectors of syntactic features derived from their parse trees T_A and T_B.

$$A : B \triangleq \overrightarrow{T_A} - \overrightarrow{T_B} = \begin{pmatrix} |T_A|_{VBD \to PRP} - |T_B|_{VBD \to PRP} \\ |T_A|_{VBD \to NNP} - |T_B|_{VBD \to NNP} \\ \vdots \\ |T_A|_{VBD \to .} - |T_B|_{VBD \to .} \end{pmatrix} \tag{3}$$

3.3. Conformity of Ratios between Trees

An analogy $A : B :: C : D$ is satisfied by checking the equality of ratios. Formula (4) defines it. For the computation of ratios between vectors and for checking for equality of ratios, we rely

[1] spaCy: https://spacy.io/

on the Python library `Nlg`[2] [22]. In this way, we extract all analogies between all parse trees corresponding to all sentences contained in our corpus.

$$A : B :: C : D \quad \overset{\Delta}{\iff} \quad \overrightarrow{T_A} - \overrightarrow{T_B} = \overrightarrow{T_C} - \overrightarrow{T_D} \tag{4}$$

3.4. Analogical Clusters

An analogical cluster is defined as a set of pairs of sentences with exactly the same ratio [23] (see definition in Formula (5)). The Python library `Nlg` can be used to extract all analogical clusters from a set of objects represented by feature vectors. We apply it for the extraction of analogical clusters between sentences, at the level of syntax. The larger an analogical cluster, the more regular the transformations between the sentences in the clusters.

$$\begin{array}{l} A_1 : B_1 \\ A_2 : B_2 \\ \quad \vdots \\ A_n : B_n \end{array} \quad \overset{\Delta}{\iff} \quad \forall (i,j) \in \{1, \ldots, n\}^2, \;\; A_i : B_i :: A_j : B_j \tag{5}$$

4. Experiments and Results

4.1. Data Used

We use the Tatoeba[3] corpus. It is a collection of sentences in more than 100 languages. Here, we use five language parts from the Tatoeba corpus: English, French, German, Polish and Finnish. The sentences we used are aligned across all five languages, they are parallel sentences that correspond to each other. Table 2 gives some statistics on this corpus. For each language, we have around eight thousand sentences. English has the lowest number of types and Finnish has the largest one among the five languages. Hapaxes are words that appear only once in a corpus. Here, we observe that English has the smallest number of hapaxes with less than 60% while Finnish has the highest percentage with over than 70%. We verify again that languages with higher morphological richness tend to have a higher number of types and hapaxes. In interest to us is the conjecture that we should extract more analogies from a language with a higher Type-Token Ratio (TTR) since type-token ratio measures lexical richness.

4.2. Metrics

To evaluate the number of analogies between sentences contained in a corpus, two metrics used in [24] are considered.

4.2.1. Analogical Density

Formula (6) defines *analogical density* as the ratio of the number of actual analogies N_{nlg} and N_s^4. If the total number of sentences in the corpus is N_s, N_s^4 is the number of possibilities of

[2] `Nlg`: http://lepage-lab.ips.waseda.ac.jp/en/projects/kakenhi-15k00317/
[3] Tatoeba: https://tatoeba.org/

Table 2
Statistics on Tatoeba corpus

Language	Number of			Average length of		TTR	Hapaxes (%)
	lines	tokens	types	token	type		
en	7,964	40,493	6,839	4.23±2.21	6.48±2.24	0.17	58.47
fr	7,964	43,563	8,581	4.62±2.71	7.46±2.58	0.20	64.10
de	7,964	41,017	8,673	4.96±2.56	7.55±2.93	0.21	63.07
pl	7,964	32,816	10,956	5.44±2.81	7.47±2.48	0.33	70.70
fi	7,964	31,152	11,270	6.09±2.95	8.10±2.90	0.36	72.17

filling in the analogy pattern with any four sentences (with possible repetition) from the corpus. As there are 8 equivalent forms of analogies [25], this should be divided by 8 to consider only individual analogies. Because the denominator is a power of 4, values for density are usually numbers of the order of 10^{-9} or 10^{-12}.

$$D_{\text{nlg}} = \frac{N_{\text{nlg}}}{\frac{1}{8} \times N_s^4} \tag{6}$$

4.2.2. Proportion of Sentences Appearing in Analogies

Formula (7) calculates the proportion of sentences appearing in analogies by dividing the number of sentences appearing in at least one analogy (N_{s_nlg}) by the total number of sentences in the corpus (N_s). This makes a percentage.

$$P = \frac{N_{\text{s\_nlg}}}{N_s} \tag{7}$$

4.3. Results and Analysis

We carry out experiments on the extraction of analogies between sentences both on the level of surface form and syntax. On the level of form, each sentence is tokenised using two different tokenisation schemes: character or word. On the level of syntax, we extract sentence analogies from a corpus by using parse trees, as described in Section 3. We also concatenate formal feature vectors with syntax feature vectors to combine the two levels. We do not conduct experiments using `char` and `word` features at the same time, because they both work on the formal level.

4.3.1. Number of Analogical Clusters

Table 3 gives the number of analogical clusters extracted from our five languages based on different feature vectors. We observe that the number of analogical clusters extracted based on syntactic trees is hundreds or thousands times larger than on the level of characters or words. Analogical clusters extracted by the combination of `char` and `tree` or `word` and `tree` are of course smaller than if only one feature is considered. When using only the `tree` feature, Finnish has a significantly higher number of analogical clusters in comparison to the other

Table 3
Number of extracted analogical clusters from different features: characters (`char`), words (`word`) and syntax (`tree`)

Language	Feature used					
	`char`		✓	✓		
	`word`				✓	✓
	`tree`	✓		✓		✓
en		502,182	774	333	325	251
fr		1,712,538	546	164	290	131
de		939,892	822	424	384	288
pl		2,246,054	860	381	333	205
fi		5,510,699	692	355	332	242

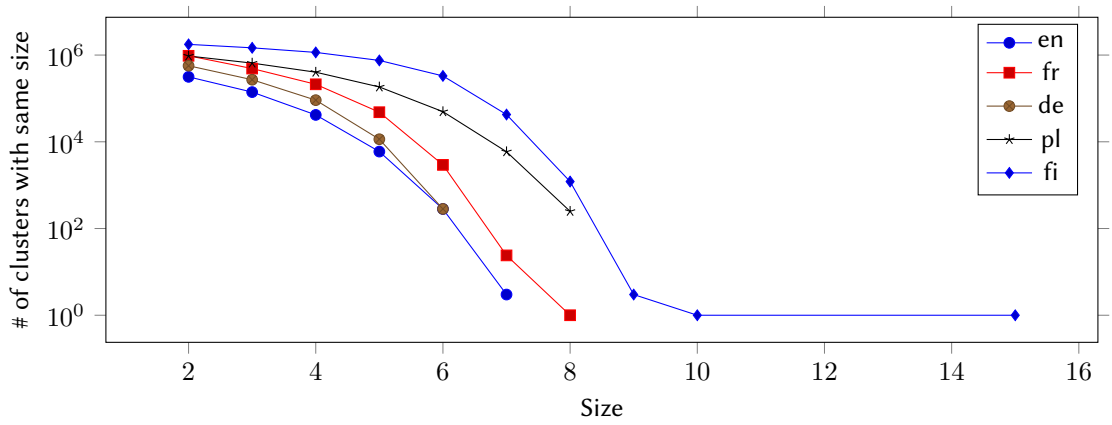

Figure 4: Number of extracted analogical clusters with the same size on the level of syntax among different languages. Caution: log scale on the y axis

languages, followed by Polish and French. Except for `char`, we observe that German always has the highest number of analogical clusters (except for `char` where it is second).

In addition, we draw the distribution of the number of analogical clusters with the same size extracted from syntactic features for our five languages in Figure 4. Although the numbers of extracted analogical clusters with the same size vary across languages, the overall trend is consistent.

4.3.2. Number of Analogies

The analogical density of the corpus is presented in Table 4. It indicates how many analogies can be extracted in the five different languages and how many sentences can be covered by these extracted analogies.

We observe that the number of analogies extracted on the level of syntax is thousands times more than on the level of form. Basically, on the level of syntax, Finnish has the highest

Table 4
Analogical densities for five different languages. The number of sentences is the number of sentences appearing in the extracted analogies. Notice the difference in orders of magnitude from `char` and `word` (10^{-12}) to `tree` (10^{-9}).

Feature	Language	Number of		Density		
		analogies	sentences	D_{nlg}		P (%)
`char`	en	985	723	1.96		9.08
	fr	679	486	1.35		6.10
	de	1,102	573	2.19	$\times 10^{-12}$	7.19
	pl	1,164	665	2.31		8.35
	fi	906	506	1.80		6.35
`word`	en	452	372	0.90		4.67
	fr	442	288	0.88		3.62
	de	603	328	1.20	$\times 10^{-12}$	4.12
	pl	559	281	1.11		3.53
	fi	580	248	1.15		3.11
`tree`	en	918,412	5,337	1.83		67.01
	fr	3,605,667	5,523	7.17		69.35
	de	1,786,002	5,336	3.56	$\times 10^{-9}$	67.00
	pl	6,369,718	6,343	12.68		79.65
	fi	20,027,663	6,999	39.83		87.88

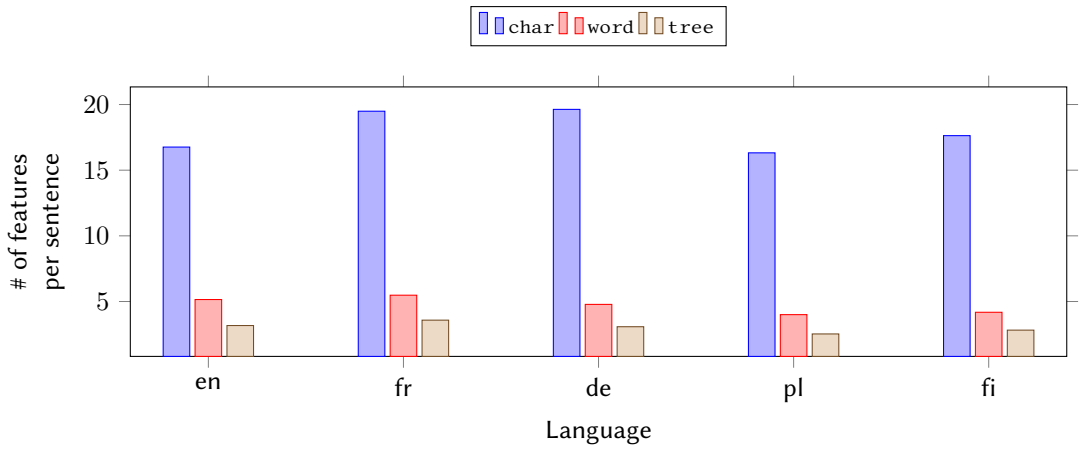

Figure 5: Number of features per sentence in extracted analogies on different levels

analogical density and the sentences that appear in analogy account for 88 percent of the whole corpus.

Figure 5 shows the number of features per sentence appearing in the extracted analogies on different levels. It is obvious that compared to the formal level, we extract more analogies on the syntactic level, given the smaller vector representations using parse trees.

Table 5
Example results of analogies in different languages extracted by combining formal and syntactic features

Lang.	Example results of sentence analogies			
en	*I bought a new car.*	: *He bought a new car.*	:: *I left the door open.*	: *He left the door open.*
fr	*Il regarde la télévision.*	: *Je regarde la télévision.*	:: *Il joue au tennis tous les jours.*	: *Je joue au tennis tous les jours.*
de	*Sie hat einen Hund.*	: *Sie hat einen Brief geschrieben.*	:: *Ich habe einen Hund.*	: *Ich habe einen Brief geschrieben.*
pl	*Jestem bohaterem.*	: *Jestem nauczycielem.*	:: *Nie jestem bohaterem.*	: *Nie jestem nauczycielem.*
fi	*Onko sinulla autoa?*	: *Onko sinulla veljiä tai siskoja?*	:: *Minulla ei ole autoa.*	: *Minulla ei ole veljiä tai siskoja.*

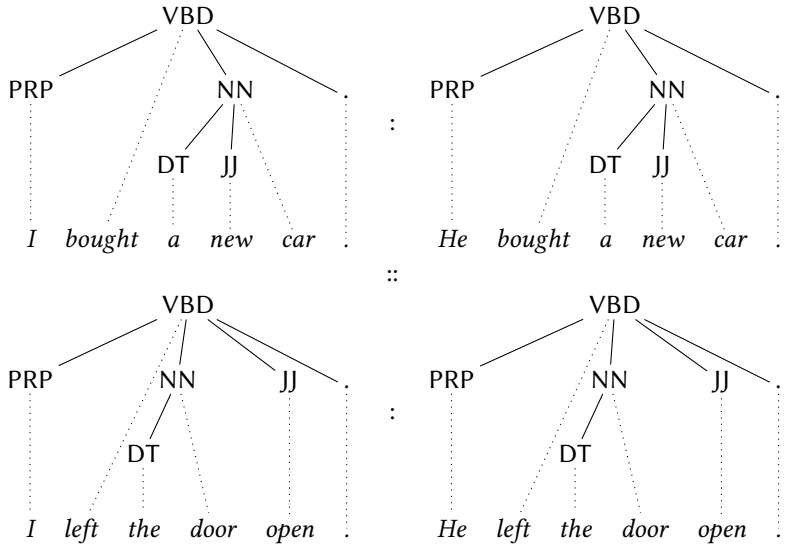

Figure 6: Parse trees of the English sentence analogy given in Table 5

4.4. Example Results of Analogies in Different Languages

In Table 5, we list some example results of sentence analogies that we extracted from the corpus in the combination of formal and syntactic features. Figure 6 plots the syntactic structure behind the first English example in Table 5.

5. Further Discussion

5.1. Analogical Grids

An analogical grid is a matrix where any four terms picked out from any two rows and any two columns is an analogy [26]. Formula (8) gives the definition of an analogical grid. The size of

We won.	:	*Tom won.*	:	*You won.*
We survived.	:	*Tom survived.*	:	*You survived.*
	:	*Tom drank too much.*	:	*You drank too much.*
We volunteered.	:	*Tom volunteered.*	:	

Figure 7: Example of an analogical grid in English extracted by combining the `tree` and `word` features

Table 6
Analogical grids extracted from different feature vectors in varying languages

Feature	Language	# of grids	Avg. size	Avg. saturation (%)
`tree` ∩ `char`	en	112	4.81	99.8
	fr	50	5.34	99.0
	de	82	6.91	98.9
	pl	94	5.72	99.4
	fi	78	5.42	99.5
`tree` ∩ `word`	en	82	4.87	99.7
	fr	39	5.51	98.7
	de	63	6.52	98.9
	pl	55	5.87	99.1
	fi	48	5.38	99.7

an analogical grid is defined as the product of its number of rows by its number of columns. As shown in Figure 7, an analogical grid may have empty cells. Thus, we can also characterise an analogical grid by the number of non-empty cells in it. This is its *saturation.* It is the ratio between the number of non-empty cells and the size of the grid.

$$\begin{array}{ccc} G_1^1 : G_1^2 : \cdots : G_1^m \\ G_2^1 : G_2^2 : \cdots : G_2^m \\ \vdots \quad \vdots \qquad\quad \vdots \\ G_n^1 : G_n^2 : \cdots : G_n^m \end{array} \quad \overset{\Delta}{\Longleftrightarrow} \quad \begin{array}{c} \forall (i,k) \in \{1,\ldots,n\}^2, \\ \forall (j,l) \in \{1,\ldots,m\}^2, \\ G_i^j : G_i^l :: G_k^j : G_k^l \end{array} \tag{8}$$

We have extracted analogical grids on the level of syntax combined with character features or word features. By extracting the analogical grids, we hope to get a more compact view of how sentences are related to each other by analogies. Based on Table 6, we observe that English has the highest number of analogical grids but also the smallest average size of analogical grids. German has the largest average size of analogical grids. The average saturation of the extracted analogical grids is all around 99 % which means the analogical grids extracted from our corpus are very dense.

5.2. Extracted analogies for different language pairs

For any language pair from the five languages in the aligned corpora, we can extract bilingual analogies by taking monolingual analogies where sentences correspond by translation. This kind of data, i.e., bilingual analogies, can then be exploited in an EBMT system by analogy.

Table 7
Number of extracted bilingual analogies for different language pairs

Feature	en-fr	en-de	en-pl	en-fi	fr-de	fr-pl	fr-fi	de-pl	de-fi	pl-fi
`char`	64	77	50	34	88	54	40	129	42	38
`tree` ∩ `char`	28	30	14	18	46	22	14	43	16	8
`word`	48	24	20	24	48	44	30	86	30	22
`tree` ∩ `word`	24	20	12	12	30	20	10	26	12	6

Table 7 counts the number of extracted bilingual analogies for different language pairs on the formal and syntactic levels. Because these are intersections, the number of bilingual analogies is of course smaller than the number of independent monolingual analogies for any of the languages in the language pair.

6. Conclusion

We proposed to extract analogies between sentences based on their syntactic structure. Experiments were carried out using Universal Dependency parse trees that allow us to compare across five different European languages. The parse trees were converted into feature vectors, the features of which were the types of branches, from which we removed the lexical information. We measured the analogical density at the syntactic level and crossed with the results at the character or word levels.

We found that the number of analogies extracted on the syntactic level is hundreds or thousands times larger than the one on the formal level, which leads to a thousand times higher analogical density. We already started extracting analogical grids to have a more compact view of how sentences are related to each other.

In this paper, we used the number of occurrences of branches in dependency representations as features to get a vector representation of sentences. Similar work could be carried out with constituency representations, if constituency parsers comparable across languages would be available. The ultimate goal of the work presented here, is to not only to extract monolingual analogies, but bilingual analogies between sentences, because they can be used by an EBMT system by analogy.

References

[1] H. Paul, Prinzipien der Sprachgeschichte, Niemayer, Tübingen, 1920.
[2] F. de Saussure, Cours de linguistique générale, Payot, Paris, 1916.
[3] L. Bloomfield, Language, Henry Holt, 1933.
[4] A. Welcomme, Hermann Paul et le concept d'analogie, CÍRCULO de Lingüística Aplicada a la Comunicación (clac) 43 (2010) 49–122.
[5] R. Fam, A. Purwarianti, Y. Lepage, Plausibility of word forms generated from analogical grids in Indonesian, in: Proceedings of the 16th International Conference on Computer Applications (ICCA–2018), Yangon, Myanmar, 2018, pp. 179–184.

[6] M. Nagao, A framework of a mechanical translation between Japanese and English by analogy principle., in: Artificial and human intelligence, 1984, p. 351–354.

[7] Y. Lepage, E. Denoual, The 'purest' EBMT system ever built: No variables, no templates, no training, examples, just examples, only examples, in: Workshop on example-based machine translation, Phuket, Thailand, 2005, pp. 81–90. URL: https://aclanthology.org/2005.mtsummit-ebmt.11.

[8] S. Dandapat, S. Morrissey, S. K. Naskar, H. Somers, Mitigating problems in analogy-based EBMT with SMT and vice versa: A case study with named entity transliteration, in: Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation, Institute of Digital Enhancement of Cognitive Processing, Waseda University, Tohoku University, Sendai, Japan, 2010, pp. 365–372. URL: https://aclanthology.org/Y10-1041.

[9] P. Langlais, Mapping source to target strings without alignment by analogical learning: A case study with transliteration, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Sofia, Bulgaria, 2013, pp. 684–689. URL: https://aclanthology.org/P13-2120.

[10] R. Rhouma, P. Langlais, Fourteen light tasks for comparing analogical and phrase-based machine translation, in: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin City University and Association for Computational Linguistics, Dublin, Ireland, 2014, pp. 444–454. URL: https://aclanthology.org/C14-1043.

[11] A. Diallo, M. Zopf, J. Fürnkranz, Learning analogy-preserving sentence embeddings for answer selection, CoRR abs/1910.05315 (2019). URL: http://arxiv.org/abs/1910.05315. arXiv:1910.05315.

[12] B. Collins, H. Somers, Recent Advances in Example-Based Machine Translation, Springer Netherlands, Dordrecht, 2003, pp. 115–153. URL: https://doi.org/10.1007/978-94-010-0181-6_4. doi:10.1007/978-94-010-0181-6_4.

[13] D. R. Hofstadter, Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought, Basic Books, Inc., USA, 1996.

[14] T. Mikolov, W.-T. Yih, G. Zweig, Linguistic regularities in continuous space word representations, in: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2013), Association for Computational Linguistics, Atlanta, Georgia, 2013, pp. 746–751. URL: http://www.aclweb.org/anthology/N13-1090.

[15] Y. Lepage, S.-i. Ando, Saussurian analogy: a theoretical account and its application, in: COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics, 1996, pp. 717–722. URL: https://aclanthology.org/C96-2121.

[16] S.-I. Ando, Y. Lepage, Linguistic structure analysis by analogy: Its efficiency, in: Proceedings of NLPRS-97, Phuket, 1997, pp. 401–406. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.50.8231. doi:10.1.1.50.8231.

[17] Y. Lepage, S.-I. Ando, S. Akamine, H. Iida, An annotated corpus in Japanese using Tesnière's structural syntax, in: Processing of Dependency-Based Grammars, 1998, pp. 109–115. URL: https://aclanthology.org/W98-0513.

[18] N. Stroppa, F. Yvon, An analogical learner for morphological analysis, in: Proceedings

of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005), Association for Computational Linguistics, Ann Arbor, Michigan, 2005, pp. 120–127. URL: https://aclanthology.org/W05-0616.

[19] N. Stroppa, F. Yvon, Analogical learning and formal proportions: Definitions and methodological issues, ENST Paris report (2005).

[20] N. Chomsky, The Logical Structure of Linguistic Theory, Springer US, 1975. URL: https://books.google.co.jp/books?id=1D66ktXOITAC.

[21] I. Chiswell, W. Hodges, Mathematical Logic, Oxford University Press, Inc., USA, 2007.

[22] R. Fam, Y. Lepage, Tools for the production of analogical grids and a resource of n-gram analogical grids in 11 languages, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Miyazaki, Japan, 2018, pp. 1060–1066. URL: https://aclanthology.org/L18-1171.

[23] Y. Lepage, C. L. Goh, Towards automatic acquisition of linguistic features, in: Proceedings of the 17th Nordic Conference of Computational Linguistics (NODALIDA 2009), Northern European Association for Language Technology (NEALT), Odense, Denmark, 2009, pp. 118–125. URL: https://aclanthology.org/W09-4618.

[24] R. Fam, Y. Lepage, A study of analogical density in various corpora at various granularity, Information 12 (2021) 314. URL: https://doi.org/10.3390/info12080314. doi:10.3390/info12080314.

[25] Y. Lepage, Languages of analogical strings, in: Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000), volume 1, Saarbrücken, 2000, pp. 488–494. URL: https://aclanthology.org/C00-1071.

[26] R. Fam, Y. Lepage, Morphological predictability of unseen words using computational analogy, CEUR Workshop Proceedings 1815 (2016) 51–60.