Analogical grids: study on morphological reinflection, lemmatisation and morphosyntactic description analysis

FAM Rashel Putraruddy Scala

October 2023

Waseda University Doctoral Dissertation

Analogical grids: study on morphological reinflection, lemmatisation and morphosyntactic description analysis

FAM Rashel Putraruddy Scala

Graduate School of Information, Production and Systems
Waseda University

October 2023

Analogical grids: study on morphological reinflection, lemmatisation and morphosyntactic description analysis

FAM Rashel Putraruddy Scala

Graduate School of Information, Production and Systems

Waseda University

A thesis submitted for the degree of

Doctor of Engineering

October, 2023

Thesis Supervisor: Professor Yves Lepage

Abstract

Out of the approximately 7,000 languages spoken worldwide, only about 20 possess text corpora consisting of hundreds of millions of words. Around 80% of the internet content is available in just 10 languages. It is common knowledge that machine learning approaches struggle when there is not enough data. Hence, recently the community of natural language processing (NLP) has become aware of the challenge of covering always more languages, particularly those with limited resources compared to English, rather than solely focusing on English. English is known to be a morphologically poor language in comparison to many languages. In a study by Bickel and Nichols (2013) on verbs, it was found that 80% of 145 sampled languages exhibit a higher degree of morphological richness than English. For instance, Basque verbs can have more than 500 different forms, whereas English typically has less than 5. Consequently, the state space of the problem is 100 times larger for Basque.

In morphologically rich languages, the problem of unseen word forms is an important issue. Unseen word forms are explainable through their relation to other words due to their morphological structure. This is a common problem for new learners of the language, and also for language models, to analyse an inflected word form, and determine the lemma and the corresponding morphosyntactic description (MSD). Reciprocally, language models need to generate the correct inflected word form from a lemma they already know. This task is even more challenging if we consider irregular forms. The implications of the above issues are significant, particularly for language learning assistance, where NLP is put in use by companies providing services in teaching languages.

This thesis relates to the automatic induction of morphology in that it first studies how word forms are organized in a language. It introduces the proposed mathematically well-defined data structure called analogical grids. We introduce a novel method to au-

tomatically extract analogical gridsfrom words contained in a corpus. Analogical grids can be seen as a step towards the automatic production of paradigm tables. Paradigm tables are produced manually by grammarians or linguists through grammatical tradition or thorough linguistic formalisation. They are known for their usefulness in learning conjugation or declension when studying a language. In a similar way to paradigm tables, analogical gridsreflect the organization of the lexicon of a language using the features used to describe the word forms. Firstly, word forms are represented as feature vectors and clusters are automatically extracted based on the ratio between word forms. Then, these clusters are automatically organized as matrices which maintain the constraint of proportional analogy between all the word forms they contain. Lastly, in application, they can be used to analyse the productivity of a language and leverage this productivity for NLP tasks. This chapter investigates the confidence in filling empty cells in analogical grids. A statistical method, Fisher's exact test, is used to measure the confidence in filling the empty cells relying on intrinsic (saturation and size) and extrinsic (word frequency and MSD) information. Experimental results show that the proposed method can generate an average of 97% of correct unseen word forms in Indonesian on the level of form, and around half on the level of form, morphology, and semantics simultaneously. Analogical grids built from the Bible corpus and SIGMORPHON 2018 Shared Task datasets are released as language resources containing more than 100 languages.

The information extracted into analogical gridsis exploited to perform two main morphological tasks: morphological generation (reinflection task) and analysis (lemmatisation and MSD analysis tasks). We carry experiments on the same dataset for both tasks.

Morphological generation task is a morphological task where given a lemma and the target MSD, generating its inflected form. This task is a standard task in the yearly evaluation campaign of SIGMORPHON Shared Task: Morphological Reinflection Task. This thesis is aligned with the current research direction and the main subject in the morphological reinflection area. Systems developed in this campaign are released publicly as available language tools. Experiments are carried out on the 2018 Shared Task which offers the largest number of languages. This allows us to evaluate the performance across many languages and against publicly available tools. We propose a holistic approach to the problem of morphological generation by treating the word form as a unit instead of breaking down word forms into smaller pieces, like morphemes, as is done is some baseline systems. The structural information and rich morphological features of word forms are used to build feature vector representations. Reinflected forms are generated by solving analogical equations between word forms encapsulated in analogical grids. We evaluate the performance of three approaches: morpheme-based (baseline system), holistic, and neural approaches. Under low-resource conditions, our proposed holistic approach improves the accuracy by 1.3% without development dataset and by 20% with the development dataset in comparison to morpheme-based approach. In addition, our holistic approach outperforms the winner system of the 2018 Shared task by 1.1% (outperforming in 60 out of 103 languages). We also found that under high-resource conditions, our holistic approach outperforms other methods with 0.1 in edit distance and outputs word forms that are closer to the answers. On average, our proposed method achieves the best performance in morphologically rich languages under low.

The morphological analysis task is the reciprocal task of morphological generation. It consists of two main subtasks, lemmatisation and MSD analysis. Our proposed method consists in lemmatizing inflected forms by solving analogical equations between the given inflected forms and word forms contained in analogical grids automatically built from the dataset. Candidates are ranked using heuristic features, such as the longest common suffix, the longest common prefix, edit distance, etc. MSD analysis is performed in the same manner by relying on morphological features instead. We compare the performance of morpheme-based, holistic, and neural approaches. Experiments are carried out on the same dataset as used at morphological generation. We compare the performance of morpheme-based, holistic, and neural approaches. Since this task does not exist in the SIGMORPHON campaign, there is no system from outside to be compared with. Under low-resource conditions (100 training instances), we found that neural approaches were considerably inferior. The results show that our holistic approach outperforms the morpheme-based approach for MSD analysis and is slightly behind for lemmatisa-

tion. However, with the development dataset, our holistic approach outperforms the morpheme-based approach by 1.3% in accuracy for lemmatisation and by 0.02 in F1 score for MSD analysis.

The main contribution in this dissertation is the novel concept of analogical grids (with a publicly released implementation). A language resource in the form of a dataset of analogical grids extracted from the SIGMORPHON 2018 Shared Task dataset containing more than 100 languages is also released. Under low-resource conditions, our proposed concept of analogical grids and our proposed holistic methods lead to an increase in performance for both the morphological generation and analysis tasks. On average, our proposed method always outperforms neural approaches under low-resource conditions (up to 3 times better performance). The proposed method is a lazy-learning method which results in a more efficient approach towards storage (no model to be saved) and time (no need for training phase).

Keywords: lexicon organization, proportional analogy, analogical grids, morphological generation, morphological analysis.

Acknowledgements

I would like to express my gratitude to all those who helped me during the writing of this thesis.

I would like to express my deepest gratitude to my supervisor, Professor Yves Lepage, for his guidance and support. His value and enthusiasm for research and teaching have motivated me to pursue excellence in my work. It is an honour to be his student and learn a lot of things during my work in his laboratory.

I would like to thank my parents for their continuous support, Mai for her love and understanding, and Noelle for the energy she gave when I needed it the most.

I would also like to express my gratitude to Professor Osamu Yoshie and Professor Mizuho Iwaihara for agreeing to serve as members of my doctoral dissertation committee and their invaluable advice on this work.

I would like to thank Wang Weihua from Inner Mongolia University for his help and collaboration on the morphological analysis task, Professor Ayu Purwarianti from Institut Teknologi Bandung for the collaboration on the experiments on explaining unseen word forms, and the late Professor Mirna Adriani from Universitas Indonesia who was the first to recommend me to pursue the doctoral degree.

I would like to thank all members of the EBMT/NLP Laboratory. I wish a great future for all of you and hope that our fate may cross again in the future.

Parts of the work reported in this thesis were supported by followings[1].

- 文部科学省:MEXT 国費外国人留学生（博士）
- JSPS Grant, Number 15K00317 (Kakenhi Kiban C), entitled "*Language productivity: efficient extraction of productive analogical clusters and their evaluation using statistical machine translation*".
- JSPS Grant, Number 18K11447 (Kakenhi Kiban C), entitled "*Self-explainable and fast-to-train example-based machine translation*".
- JSPS Grant, Number 21K12038 (Kakenhi Kiban C), entitled "*Theoretically founded algorithms for the automatic production of analogy tests in NLP*".

[1]**Disclosure statement**: The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

"Soli Deo Gloria"

List of abbreviations

ACL Association for Computational Linguistics

FIFO First-In-First-Out

GRU Gated Recurrent Unit

LCS Longest common sub-sequence

LLM Large language model

LSTM Long Short-Term Memory

MSD Morphosyntactic description

MSF Morphosyntactic feature

NLP Natural language processing

OOV Out-of-vocabulary

PCA Principal Component Analysis

SIGMORPHON Special Interest Group on Association for Computational Morphology and Phonology

seq2seq Sequence-to-sequence

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter introduces the background and motivation of the research. Based on the motivation of this work, the contributions made by this research are summarised in the following section.

1.1 Background

In this section, we provide the background of the research. We also present several approaches to the problem of explaining unseen words, particularly the inflected ones.

1.1.1 Explaining unseen words

In recent years, NLP has been dominated by the paradigm of extracting knowledge from a training corpus, which is then applied to perform a given task on a test set, with the system's performance evaluated accordingly. Since many techniques have been initially developed for English, the typographic word is often adopted as the fundamental processing unit in NLP tasks such as machine translation and speech recognition. The vocabulary of the system is the collection of the words that are known to the system, while the presence of unseen words, OOV or new words, poses a significant challenge. In fact, these unseen words share similarities with hapaxes, which are estimated to account for 30% to 50% of the vocabulary in any typical English text, with 44% observed in Part A of the British National Corpus. Despite their rarity, hapaxes represent a substantial proportion of the vocabulary, constituting less than 0.2% of the total word count in the same corpus.

NLP systems are expected to analyse previously unseen words contained in a given text by identifying the lemma and morphosyntactic descriptions.

This is a morphological analysis task. Reciprocally, these systems are required to generate word forms from lemmata and the target morphosyntactic descriptions. This is a morphological generation task.

1.1.2 Automatic induction of morphology

The syntactic and semantic relation is reflected by the word's form relative to its other form. As an example, English has singular and plural forms for its nouns, e.g. *table* is to *tables* as *chair* is to *chairs*. This is a phenomenon of morphological inflection in English. We believe that unseen words can be explained by exploiting this relation between the word forms, and how we derived a word form from another one.

When automatically inducing morphology, several choices are made. The first one is fundamental and creates debate: whether words should be explicitly decomposed or not. The most common approach seems to be to decompose words into components, i.e., to adopt a morpheme-based morphology point of view. This approach is encouraged in the Morpho Challenge evaluation campaign described as "unsupervised morpheme analysis" in (Kurimo et al., 2010). The alternative is to avoid decomposition explicitly and to consider the word as a basic unit. This approach is referred to as the lexeme-based morphology approach. E.g., Anderson (1992) puts forward such a theoretical model, while Hathout (2009) reports practical experiments in which morphological families and derivational series are acquired automatically without explicit decomposition. The model combines graph-representations of the lexicon and proportional analogies between words (Lepage, 2004). The graph representations of the lexicon are explored through random walks (Gaume et al., 2006; Muller et al., 2006).

The second choice is usually a technical choice of what kind of phenomena should be captured, among all possible affixing combinations, reduplications, gemination, etc. encountered in word formation. The names describing the phenomena reflect the point of view adopted: phonological (e.g., speaking of alternation), morphological/grammatical (e.g., speaking of apophony) (Kuryłowicz, 1961) or simply formal (e.g., speaking of substitution). To illustrate with works that use an explicit decomposition of word forms and that describe the phenomena formally, one finds, for example, works that deal only with suffixing (Wegari et al., 2015), or with both prefixing and suffixing (Soricut and Och, 2015), or yet take into account infixing in addition to prefixing and suffixing (Gasser, 2009, 2011).

A third choice concerns the status of semantics in the task of induction. Semantics may be taken into account from the beginning, as input or during morphological induction, or discovered afterwards in support of the structure of the lexicon obtained, or imposed over the results of automatic induction to filter results. As possible examples, Soricut and Och (2015) use the Skip-Gram model (Mikolov et al., 2013a,b) to obtain word vector representations on which they discover regular prefixes and suffixes; (see also (Levy and Goldberg, 2014) and (Pennington et al., 2014)). On the contrary, Luong et al. (2013) or Botha and Blunsom (2014) employ external morphological analyzers, such as Morfessor (Creutz and Lagus, 2007), to perform morpheme segmentation and morphological analysis, and combine the results with vector representations afterwards.

1.1.3 Approaches to the morphological task

Many NLP tasks, like machine translation, require analysis and generation of morphological word forms, even previously unseen ones. Different languages exhibit different levels of richness in morphology. This makes the task an interesting problem. Dryer and Eisner (2011) show that data sparsity is a common issue for languages with rich morphology, which usually leads to poor generalisations in machine learning. There are currently three main approaches to the problem.

The hand-engineered rule-based approach offers high accuracy but it is time-consuming during the construction phase. It usually faces the word coverage problem and is usually language-dependent.

The supervised approach automatically induces rules from a given training data and applies the best rules to generate the target forms by using some classification techniques (Ahlberg et al., 2015). It is practically language-independent and relatively easier to build. However, data sparsity is still an issue.

The neural approach is the model that triumphed in the task recently, especially the RNN encoder-decoder model (Kann and Schütze, 2016; Makarov et al., 2017). Some drawbacks of this approach are long training times and the need for large amounts of training data. It is common knowledge that the neural approach suffers from a lack of training data.

1.2 Motivation

Keeping in mind the drawbacks of approaches mentioned in the previous section, we are interested in developing a system that is language-independent but also relatively easy to build. We can summarise that the issues of the previous approaches are:

- **Time**: This concerns the time needed to train the systems or manual construction of rules.
- **Size**: This concerns the efficiency of the size of trained models at the solving task. Another thing is the size of the data that is needed to build the system. For this, we are more interested in low-resourced situations in comparison to high-resourced situations.

We consider computational analogy as a possible way of generating and explaining unseen words. We propose a novel concept of analogical grids along with a pipeline to automatically produce analogical grids from a given set of words.

1.2.1 High vs. low-resourced languages

Recently in NLP, there has been a true concern covering all languages, especially English. For languages that are already studied, such as English, there are many resources available. As NLP is not a synonym for English (Bender's rule), there are more languages emerging to be tackled. These languages have significantly fewer resources in comparison to English. Only around 20 out of around 7,000 languages spoken in the world have text corpora of hundreds of millions of words. As a practical issue, it is estimated that 80% of content on the internet is available in only one of 10 languages[1]. It is common knowledge that machine learning approaches struggle when there is not enough data.

Figure 1.1 provides an estimation of available NLP tools and resources by language. Out of all NLP solutions developed, more than two-thirds of them are for English despite sharing only 10% of the total number of speakers. In comparison to that, the number of NLP solutions for low-resource languages is over 11 times smaller than English while having almost 7 times more people speaking the language. These languages are mostly spoken in Africa and Asia.

[1] https://www.consumersinternational.org

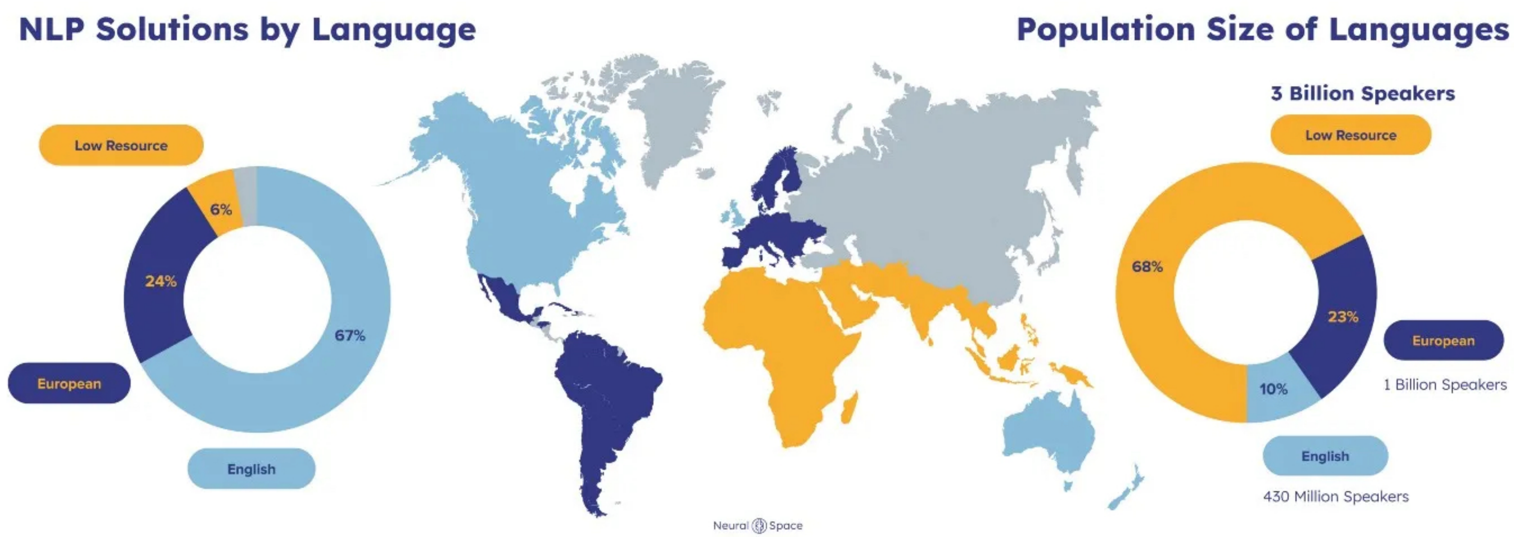


Figure 1.1: Estimation of language resources and tools available for languages in comparison to the number of speakers of the language. (Figure copied from `https://medium.com/neuralspace/`)

Moving on to the morphological richness of languages, English is known to be a morphologically poor language in comparison to other languages. Bickel and Nichols (2013) reported a study on the morphological richness of verbs in different languages. The study shows that 80% of 145 sampled languages have a higher degree of morphological richness than English. For example, a verb in Basque may have more than 500 different forms in comparison to English with only less than 5. The size of the state space of the problem is 100 times larger for Basque.

1.2.2 Natural language processing research in the era of emerging large language models

Dissemination of large language models (LLM), such as ChatGPT[1] by OpenAI and Bard[2] by Google, has taken a lot of attention from the world to NLP research more than ever before. These systems are claimed to be general-purpose language models disseminated as chat-bots available for public use. There are discussions on the impact of these systems on the scientific methodology and whether these systems should be taken into account as baselines when comparing the results of experiments in NLP.

Let us review again the criteria of a good baseline[3].

1. **Open**: The code, data, and documentation are available to be downloaded.

2. **Reasonably reproducible**: There is enough information available to reproduce the system/model using the provided code, data, and documentation.

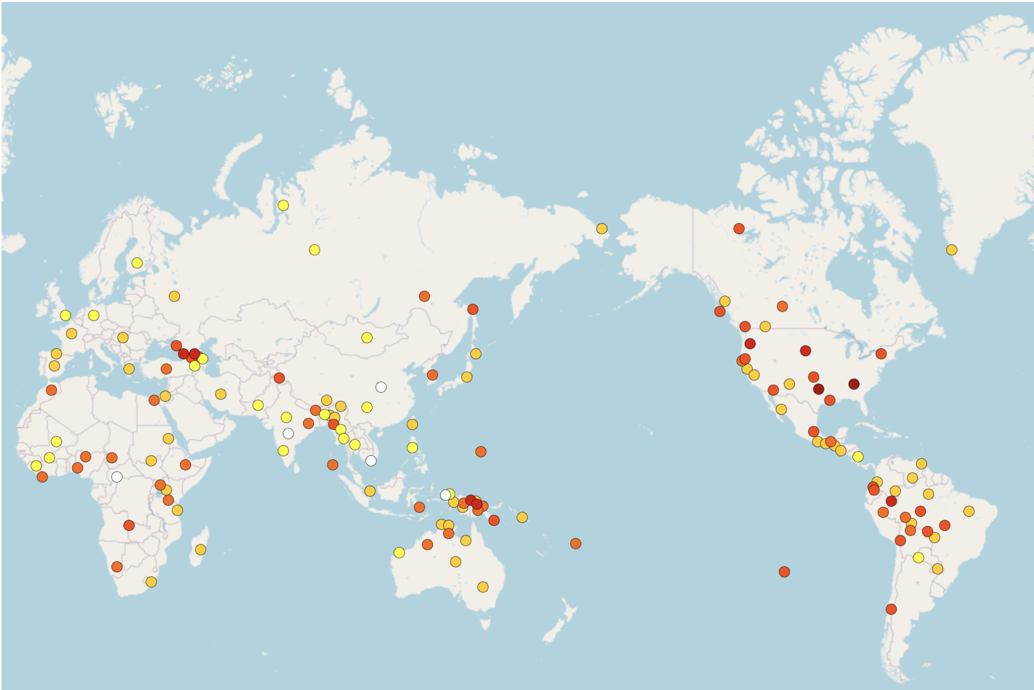
However, these LLMs are considered closed and not reasonably reproducible models. First, there is no documentation on what data is used and on what kind of architecture the model was built. Second, it was stated in the technical report[4] by OpenAI that there was a data contamination problem during training. Thus, there is an unclear test-train overlap issue that breaks a fundamental research methodology for carrying out experiments. These models can be considered an important oracle but cannot, by any means, be used as a point of comparison. For these reasons, these LLMs cannot be meaningfully studied and considered as a requisite baseline.

[1] `chat.openai.com`

[2] `bard.google.com`

[3] `https://hackingsemantics.xyz/2023/closed-baselines/`

[4] `https://cdn.openai.com/papers/gpt-4.pdf`



	Inflectional Synthesis of the Verb	Number of languages
+	0-1 category per word	5
+	2-3 categories per word	24
+	4-5 categories per word	52
+	6-7 categories per word	31
+	8-9 categories per word	24
+	10-11 categories per word	7
+	12-13 categories per word	2

Figure 1.2: A map of languages with the number of categories per verb in the language (*top*) and the statistics of the number of languages against the number of categories per word (*bottom*). (Figure copied from `https://wals.info/combinations/22A?`)

As for the language representation issue, these models are mainly available only for high-resourced languages. Bard is available only in English. This is intuitive considering that GPT-4, the architecture that these models are based on, needs a lot of training data.

To close this section, below are quotes from some leading scientists in the NLP field on the issue of closed models. Our work in this thesis aligns with these ideas.

> "Though English and Mandarin Chinese are widely spoken, both as first and second languages, clearly **a world in which advanced language technology exists only for these two languages is undesirable.**" - *The #Bender Rule: On Naming the Languages We Study and Why It Matters* by Emily M. Bender (Executive Board of the Association for Computational Linguistics, Professor in the Department of Linguistics at the University of Washington)

> "We are NLP researchers, and at the absolute minimum our job is to preserve the fundamentals of scientific methodology. . . . Does it work well? Yes. Is it a magical "emergent" property? No. Can we develop another paraphrasing system and meaningfully compare it to this one? Also no. And this is where it stops being relevant for NLP research. **That which is not open and reasonably reproducible cannot be considered a requisite baseline.** " - *Closed AI Models Make Bad Baselines* by Anna Rogers (Co-program chair of Association for Computational Linguistics 2023, Assistant Professor in Computer Science Department at the IT University of Copenhagen)

> "I do not expect Coca-Cola to present its secret formula. But nor do I plan to give them **scientific credibility** for alleged advances that we know nothing about." - *The Sparks of AGI? Or the End of Science?* by Gary Marcus (Leading scientist in artificial intelligence field, Emeritus Professor of Psychology and Neural Science at New York University)

1.3 Contributions

The contributions of this thesis can be summarised as follows.

1. A novel concept of analogical grids (its implementation as a Python module and public release).

2. A study of the application to morphological tasks: morphological generation and morphological analysis (lemmatisation and morphosyntactic analysis).

3. The release of language resources in the form ofan analogy dataset extracted from the SIGMORPHON 2018 Shared Task dataset which contains more than 100 languages.

1.4 Organisation of the thesis

This thesis is organised as follows. The methodology to automatically extract analogical clusters from a given text and construct them into analogical grids is explained in Chapter 2. Chapter 3 presents the application of our notion about analogical grids to the morphological generation task. The application to the morphological analysis task is introduced in Chapter 4. Both chapters present the experiments and analysis of the results. There will be further discussion regarding the language complexity and issue about data sizes at the end of both chapters. Chapter 5 gives the conclusion and directions for future works. Figure 1.3 shows the overall organisation of the thesis and the connection between chapters.

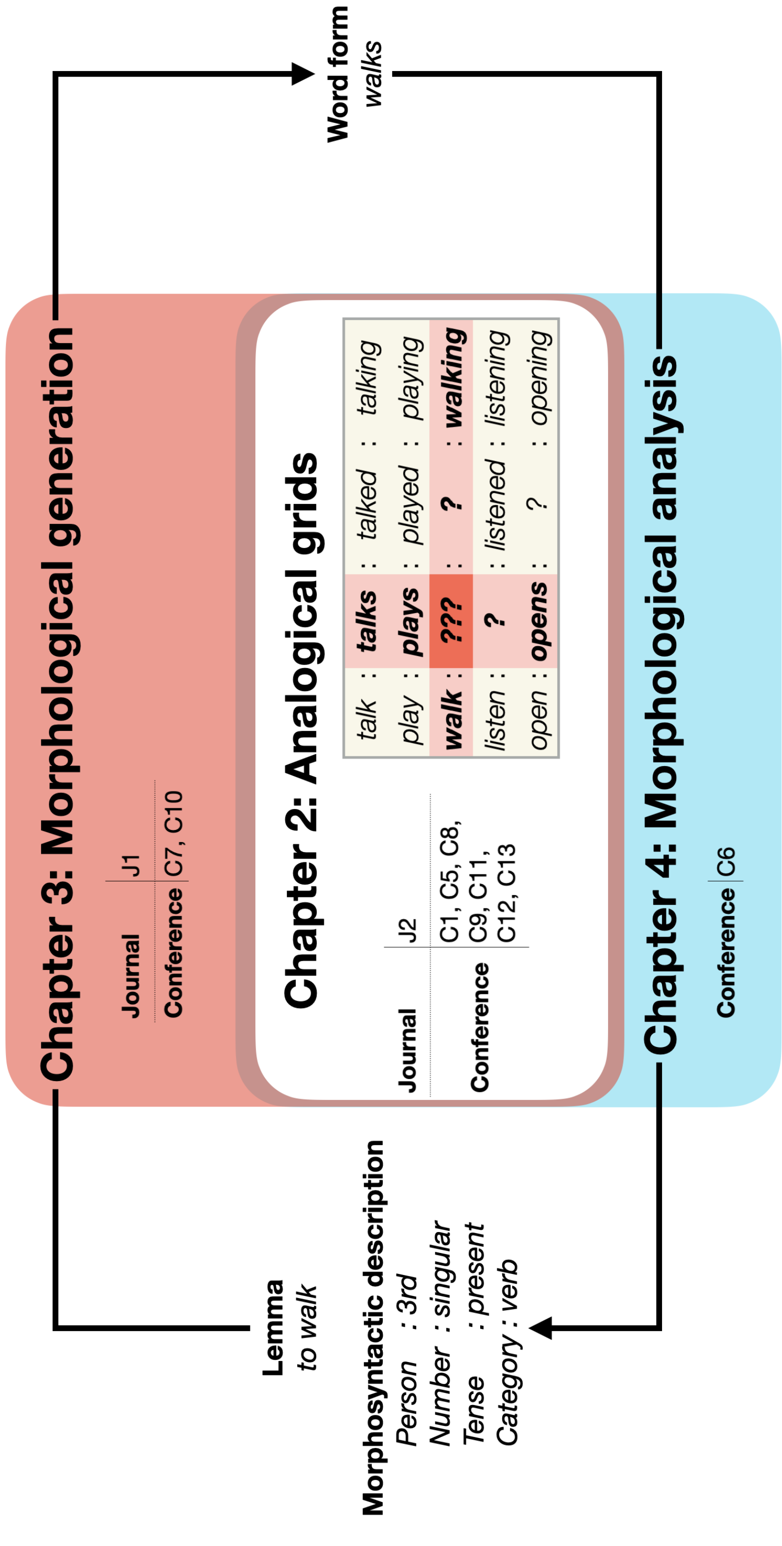


Figure 1.3: Organisation of the thesis

Chapter 2

Analogical grids: automatic organisation of a lexicon

This chapter introduces the proposed mathematically well-defined data structure called analogical grids. We introduce a novel method to automatically extract analogical grids from words contained in a corpus. Experimental studies on the saturation and size of analogical grids are carried out in several languages.

2.1 Organisation of the chapter

This chapter is organised as follows: Section 2.3 introduces the novel concept of analogical grids. A pipeline to produce analogical grids from a set of words is also described. Section 2.4 presents a study on the two properties of analogical grids: size and saturation. Preliminary experiments are carried out on filling out empty cells in analogical grids. Section 2.6 gives the summary of the chapter.

2.2 List of publications

The research described in this chapter has been published in the following publications[1].

Journal paper

(J2) Fam, R. and Lepage, Y. (2021). A study of analogical density in various corpora at various granularity. *Information*, 12(8)

Conference paper with reviewing committee

(C1) Fam, R. and Lepage, Y. (2023a). Investigating parallelograms: Assessing several word embedding spaces against various analogy test sets in several languages using approximation. In *Proceedings of the 10th Language and Technology Conference (LTC–2023)*, pages 68–72, Poznań, Poland. Fundacja uniwersytetu im. Adama Mickiewicza

(C5) Fam, R. and Lepage, Y. (2019). A study of analogical grids extracted using feature vectors on varying vocabulary sizes in Indonesian. In *Proceedings of 2019 International Conference on Advanced Computer Science and Information Systems (ICACSIS–19)*, pages 255–260, Bali, Indonesia

(C8) Fam, R. and Lepage, Y. (2018b). Tools for The Production of Analogical Grids and a Resource of N-gram Analogical Grids in 11 Languages. In chair), N. C. C., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Hasida, K., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S., and Tokunaga, T., editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC–2018)*, Miyazaki, Japan. European Language Resources Association (ELRA)

(C9) Fam, R., Purwarianti, A., and Lepage, Y. (2018). Plausibility of word forms generated from analogical grids in Indonesian. In *Proceedings of the 16th International Conference on Computer Applications (ICCA–2018)*, pages 179–184, Yangon, Myanmar. UCSY

(C11) Fam, R. and Lepage, Y. (2017b). A study of the saturation of analogical grids agnostically extracted from texts. In *Proceedings of the Computational Analogy Workshop at the 25th International Conference on*

[1]Numbering follows the document *04-Research achievements publications* submitted together for the degree application.

Case-Based Reasoning (ICCBR-CA–2017), pages 11–20, Trondheim, Norway

(C12) Fam, R., Lepage, Y., Gojali, S., and Purwarianti, A. (2017b). A study of explaining unseen words in Indonesian using analogical clusters. In *Proceedings of the 15th International Conference on Computer Applications (ICCA–2017)*, pages 416–421, Yangon, Myanmar

(C13) Fam, R. and Lepage, Y. (2016b). Morphological predictability of unseen words using computational analogy. In *Proceedings of the Computational Analogy Workshop at the 24th International Conference on Case-Based Reasoning (ICCBR-CA–2016)*, pages 51–60, Atlanta, Georgia

> *Anto memakan nasi dan meminum air. Nasi itu dibeli di pasar. Di pasar, Anto melihat mainan. Anto senang main bola. Setelah main, Anto suka minum es dan makan cilok. Makanan dan minuman itu juga dia beli di pasar. Es dan cilok memang enak dimakan dan diminum selesai olahraga.*

air anto ***beli*** *bola cilok dan di dia* ***dibeli dimakan diminum*** *enak es itu juga* ***main mainan makan makanan*** *melihat* ***memakan*** *memang* ***meminum minum minuman*** *nasi olahraga pasar selesai senang setelah suka*

Figure 2.1: A text in Indonesian (*above*) and the list of words extracted from it (*below*). Words appearing in the next figures are boldfaced.

2.3 Automatic organisation of lexica into analogical grids

In this section, we present basic notions related to analogical grids. Analogical grids can be seen as a step towards the automatic production of paradigm tables. Paradigm tables are produced manually by grammarians or linguists through grammatical tradition or thorough linguistic formalisation. They are known for their usefulness in learning conjugation or declension when studying a language. In a similar way to paradigm tables, analogical grids reflect the organisation of the lexicon of a language using the features used to describe the word forms.

A pipeline to extract analogical grids is also introduced. The following pipeline relies on the notion of computational analogy between strings of symbols proposed in (Lepage, 2004). Firstly, word forms are represented as feature vectors and clusters are automatically extracted based on the ratio between word forms. Then, these clusters are automatically organised as matrices which maintain the constraint of proportional analogy between all the word forms they contain. Lastly, in application, they can be used to analyse the productivity of a language and leverage this productivity for NLP tasks.

2.3.1 Ratio between words

The top of Figure 2.1 is a forged example text in Indonesian, a language that is known for its relative richness in derivational morphology. We intentionally do not give its translation into English to place the reader in the agnostic position of the computer in front of such data. The list of words, sorted in lexicographic order, that can be extracted from this text, is given at the bottom of Figure 2.1.

From this word list, some commonalities between words can be identified at a glance. Some words can be viewed as sharing some common part. An example is the word *makan* and the word *dimakan*. Another is the words *bola* and *beli* which share the same consonants in the same order: *b* and *l*.

$$A : B \quad \triangleq \quad \begin{pmatrix} |A|_a - |B|_a \\ |A|_b - |B|_b \\ \vdots \\ |A|_z - |B|_z \\ \mathrm{d}(A, B) \end{pmatrix} \qquad \mathit{makan} : \mathit{makanan} \triangleq \begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \\ 2 \end{pmatrix} \tag{2.1}$$

The existence of only one pair is not enough to support the evidence that two words are actually in relation to one with the other. In the case of *bola* and *beli*, no other pair can be found in the list of words, so that we cannot conclude whether this reflects some phenomenon in the Indonesian language. Evidence is missing. On the contrary, for the words *makan* and the word *makanan*, the same *ratio* is seen to hold between several other word pairs from the same text, like *minum* and *minuman*, or *main* and *mainan*. As a matter of fact, Indonesian morphology tells that *makanan* 'food' is derived from *makan* 'to eat' by using the suffix *-an* which builds a noun from an active verb. The other words can be translated into English as 'to drink', 'beverage', 'to play' and 'toy' respectively.

We first define the ratio between two words A and B as a vector of features made of all the differences in number of occurrences in the two words, for all the characters, whatever the writing system; plus the distance between the two words. This is taken from the characterizations of the proportional analogy of commutation (Lepage, 2004; Stroppa and Yvon, 2005; Langlais and Yvon, 2008). The only two edit operations involved are insertion and deletion[1]. See definition in ((2.1)).

The notation $|S|_c$ stands for the number of occurrences of character c in string S and $d(A, B)$, which is the edit distance between two strings A and B. The above definition of ratios captures prefixing and suffixing. Although we do not show it here, this definition also captures parallel infixing or

[1]The purpose is to indirectly take into account the number of common characters appearing in the same order in A and B because $d(A, B) = |A| + |B| - 2 \times s(A, B)$ where $|S|$ denotes the length of string S and $s(A, B)$ the length of the longest common sub-sequence (LCS) between A and B.

Table 2.1: Examples of analogies in different languages illustrating different phenomena. The formalisation used in this work captures infixing, but not repetition and reduplication

Phenomenon	Language	Example
Repetition	Indonesian	*pasar* : *pasar-pasar* :: *kota* : *kota-kota* 'market' : 'markets' :: 'town' : 'towns'
Reduplication	Latin	*cado* : *cecidi* :: *pago* : *pepigi* 'I fall' : 'I fell' :: 'I conclude' : 'I concluded'
Infixing	Arabic	*kalb* : *kulaib* :: *masjid* : *musaijid* 'a dog' : 'dogs' :: 'a mosque' : 'mosques'

interdigitation, a well-known phenomenon in the morphology of Semitic languages (Beesley, 1998; Wintner, 2014). However, partial reduplication (e.g. consonant spreading) or total reduplication (Gil, 2002) (e.g., marked plural in Indonesian) are not captured by this definition. Examples of different phenomena are listed in Table 2.1.

2.3.2 Extracting analogical clusters

Based on the notion of ratio, we then define an analogy, more precisely a proportional analogy of commutation between strings of symbols, as a relationship between four objects where two properties are met:

- equality of ratios between the first and the second terms on one hand and the third and the fourth terms on the other hand, and
- exchange of the means.

The exchange of the means states that the second and the third terms can be exchanged. The notation and the definition of an analogy are given in ((2.2)) at the same time[1] (Lepage, 1998; Langlais and Yvon, 2008; Stroppa and Yvon, 2005).

$$A : B :: C : D \quad \stackrel{\Delta}{\Longleftrightarrow} \quad \begin{cases} A : B &= C : D \\ A : C &= B : D \end{cases} \tag{2.2}$$

[1] Trivially, $|A|_a - |B|_a = |C|_a - |D|_a \quad \Leftrightarrow \quad |A|_a - |C|_a = |B|_a - |D|_a$. Hence, the equalities on features added by $A : C = B : D$ in ((2.2)) in fact reduce to one: $d(A, C) = d(B, D)$.

From the entire set of words contained in a text, we compute the set of analogical clusters, i.e., a series of word pairs in which any two word pairs is a proportional analogy as defined in ((2.2)). Such analogical clusters are defined in ((2.3)). Notice that the order of word pairs in analogical clusters has no importance.

$$\begin{array}{c} A_1 : B_1 \\ A_2 : B_2 \\ \vdots \\ A_n : B_n \end{array} \quad \stackrel{\Delta}{\iff} \quad \forall (i,j) \in \{1,\ldots,n\}^2, \quad A_i : B_i :: A_j : B_j \tag{2.3}$$

To produce the set of analogical clusters, we first group pairs of words by equal ratio in the number of characters using the method proposed in (Lepage, 2014). The complexity is $O(n^2)$ in the worst case with n the number of words. We then test for equality between distances for each word pair. This may split the sets of word pairs into smaller sets of word pairs for which all word pairs have the same ratio.

$$\begin{array}{r@{\,:\,}l} \textit{makan} & \textit{makanan} \\ \textit{minum} & \textit{minuman} \\ \textit{main} & \textit{mainan} \end{array} \qquad \begin{array}{r@{\,:\,}l} \textit{minum} & \textit{diminum} \\ \textit{makan} & \textit{dimakan} \\ \textit{beli} & \textit{dibeli} \end{array}$$

$$\begin{array}{r@{\,:\,}l} \textit{minum} & \textit{meminum} \\ \textit{makan} & \textit{memakan} \end{array} \qquad \begin{array}{r@{\,:\,}l} \textit{makan} & \textit{minum} \\ \textit{makanan} & \textit{minuman} \\ \textit{dimakan} & \textit{diminum} \\ \textit{memakan} & \textit{meminum} \end{array}$$

Figure 2.2: Four analogical clusters of different sizes extracted from the list of words given in Figure 2.1: three word pairs for the two series *above*, two and four word pairs respectively for the two series *below*.

Finally, for each such set of word pairs with an equal ratio, we test for equality in edit distance vertically, i.e., we verify that $A_i : A_j = B_i : B_j$ for any pair of word pairs (i, j) (see Footnote 1). Cases, where the equality is not met, lead to split the set into smaller sets. Ideally, this is equivalent to extracting all maximal cliques in the undirected graph whose set of vertices is a word pair i and where there is an edge between word pair i and word pair j if and only if the constraint $A_i : A_j = B_i : B_j$ is met. Existing algorithms for this problem (Bron and Kerbosch, 1973) are time-consuming. For this reason, we adopt a heuristic that does not ensure that all maximal cliques are output but ensures that all nodes belong to one of the maximal cliques output (see Algorithm 1 in Appendix C). We ensure that any two word pairs in a series of word pairs of equal ratio, say, A, B and C, D, also verifies $A : C = B : D$.

Practically, it would be too long to compute all possible ratios between all pairs of words directly, so a strategy in two steps is adopted following a method proposed in (Lepage, 2014). Analogical clusters have been used between sentences (Wang et al., 2014) or between Chinese characters (Lepage, 2014).

2.3.3 Producing analogical grids

Individual analogical clusters already give some insight at the organisation of the lexicon. Analogical grids (Singh and Ford, 2000; Neuvel and Fulop, 2002; Hathout, 2008) give a more compact view by merging several analogical clusters. An analogical grid is a matrix of words where four words from two rows and two columns are an analogy ((2.4)). As the order of rows and columns is indeed not relevant, one should think of a torus in three-dimensional space, rather than a matrix in two dimensions.

$$\begin{array}{l} G_1^1 : G_1^2 : \cdots : G_1^m \\ G_2^1 : G_2^2 : \cdots : G_2^m \\ \;\vdots \quad \vdots \qquad\quad \vdots \\ G_n^1 : G_n^2 : \cdots : G_n^m \end{array} \quad \stackrel{\Delta}{\iff} \quad \begin{array}{l} \forall (i,k) \in \{1,\ldots,n\}^2, \\ \forall (j,l) \in \{1,\ldots,m\}^2, \\ \qquad G_i^j : G_i^l :: G_k^j : G_k^l \end{array} \tag{2.4}$$

The definition of analogical grids in Formula (2.4) implies that any four word forms at the intersection of two rows and two columns (G_i^j, G_i^l, G_k^j and G_k^l) make an analogy between sequences of characters.

Analogical grids can be used to study word productivity in a given language as shown in (Singh and Ford, 2000; Neuvel and Fulop, 2002; Hathout, 2008). They can also be used to make comparisons across languages as in (Fam and Lepage, 2016b), where the goal is to predict missing word forms by using neighbouring word forms inside analogical grids.

We create analogical grids from analogical clusters as follows. An analogical grid is first initialized from one analogical cluster and then expanded by adding other analogical clusters to it. There are two possible ways of adding a cluster to an analogical grid. In the first case, if a column in the analogical grid shares at least three words with a column in an analogical cluster, this cluster can be added vertically to it. In the second case, an analogical cluster

makan :	*dimakan* :	*memakan* :	*makanan*
minum :	*diminum* :	*meminum* :	*minuman*
main :	:	:	*mainan*
beli :	*dibeli* :	:	

Figure 2.3: The analogical grid obtained by application of Algorithm 2 on the set of analogical clusters given in Figure 2.2. The last series in Figure 2.2 has been inserted horizontally as the two top rows.

shares more than three words on a row of the analogical grid, so that the cluster can be transposed and inserted to the analogical grid horizontally.

Algorithm 2 sketches the necessary functions for the production of analogical grids from analogical clusters. In these functions, the strategy is to process longer analogical clusters first because the possibility of inserting smaller new series in an analogical grid increases with the number of words it contains. To ensure that no insertion is forgotten, the list of series of word pairs of equal ratio is scanned several times. The complexity is $O(n^2)$ in the worst case with n the number of clusters. However, the algorithm is implemented in a way such that the clusters are added only to one analogical grid. Thus, in practice, the complexity is sub-quadratic. See Algorithm 2 in Appendix 2 for more details.

It is worth noticing that, when creating all possible analogical grids from a text, not all of the words will necessarily appear in an analogical grid. Reciprocally, analogical grids extracted from texts may contain blank cells. An analogical grid that does not contain any blank cell is not productive as no new word can be entered into it. On the contrary, we will call any analogical grid that contains at least one blank cell a *productive analogical grid*. We will call a word that may fill a blank cell in a productive analogical grid a *predictable word*.

In the experiments reported hereafter, we monitor the density of the analogical grids produced by controlling the addition of analogical clusters: we add a cluster to an analogical grid only if the density of the new analogical grid after adding the cluster is above a given threshold. This is done by the condition in the function EXPAND_TABLE in Algorithm 2.

2.3.4 Inequality between ratios: enforcement by analogical grids

An apparent defect of the previous definition of word ratio in Formula (2.1) is that it gives the same vector for the ratios:

$$\begin{array}{ccccccc} \textit{makan} : \textit{makanan} & & \textit{main} : \textit{mainan} & & \textit{makan} : \textit{main} & & \textit{makanan} : \textit{mainan} \\ \begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \\ 2 \end{pmatrix} & = & \begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \\ 2 \end{pmatrix} & \& & \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 3 \end{pmatrix} & = & \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 3 \end{pmatrix} \end{array}$$

$$\Rightarrow$$

$$\textit{makan} : \textit{makanan} :: \textit{main} : \textit{mainan}$$

Figure 2.4: The two ratios between pairs of words *makan*, *makanan*, *main* and *mainan* contained in Figure 2.3.

- $\textit{makan} : \textit{makanan}$,
- $\textit{makan} : \textit{makaann}$ (note the *a*s and *n*s in the middle of *makaann*), and
- $\textit{makan} : \textit{makanna}$ (note the two *n*s in the middle of *makanna*).

This is due to the use of insertion and deletion as the only edit operations. The defect is eliminated by the use of analogical grids. The purpose of working with analogical grids, and not only with individual analogies, is that Formula (2.4) imposes more constraints for a word form to enter a grid: a word form in a grid must satisfy all analogy relationships with all surrounding word forms in the grid. The word form *makanan* in the analogical grid of Figure 2.3 is the only word form which fits in, among *makanan*, *makaann*, or *makanna*. For example, as proved below, using the words *main* and *mainan* from the analogical grid, the inequality of the ratios $\textit{makan} : \textit{main}$ and $\textit{makaann} : \textit{minuman}$ implies that there is no analogy between these four words. The same holds for the word form *makanna*. In all these cases, the inequality comes from different edit distance values.

$$\begin{array}{ccc} \textit{makan} : \textit{main} & & \textit{makaann} : \textit{mainan} \\ \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 3 \end{pmatrix} & \neq & \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 5 \end{pmatrix} \end{array}$$

$$\Rightarrow$$

$$\textit{makan} : \textit{main} \not:: \textit{makaann} : \textit{mainan}$$

The above discussion suggests that there should be a relationship between the size of the analogical grids and the freedom to fill an empty cell in an analogical grid.

2.4 Study on size and saturation of analogical grids

In this section, we present two main properties of analogical grids: size and saturation. We then perform a study on the relation of these two properties across languages. When analogical grids are produced from a set of words contained in a corpus, there is a high chance that there are empty cells in those analogical grids. We performed preliminary experiments in filling these empty cells as a study on language productivity.

2.4.1 Size and saturation of analogical grids

The size of an analogical grid is defined as the product of its number of rows by its number of columns, by Formula (2.5). It is the total number of cells inside an analogical grid. The two analogical grids in Figure 2.3 have sizes of $4 \times 5 = 20$ (*left*) and $4 \times 4 = 16$ (*right*) respectively.

$$\text{Size} \triangleq \text{Number of rows} \times \text{Number of columns} \tag{2.5}$$

Let us now turn to the number of empty cells of an analogical grid, or rather the number of non-empty cells which we call its *saturation*[1]. We compute it using Formula (2.6). In Figure 2.3, there are 4 empty cells. The saturation is thus: $100 - (4 \times 100)/16 = 75\,\%$.

$$\text{Saturation} \triangleq 100 - \frac{\text{Number of empty cells} \times 100}{\text{Size}} \tag{2.6}$$

In that grid, *dimain* is a candidate to fill in the empty cell on the 3rd row and 2nd column because the ratio between *dimain* and *main* is equal to the ratio of all other word pairs on the 2nd and the 1st columns and also because the ratio between *dimain* and *mainan* is equal to the ratio of all other word pairs on the 2nd and the 4th columns, and similarly for the third row and other rows. However, *dimain* is not a valid Indonesian word. *Belian* can be forged in the cell on the 4th row and the 4th column by considering all ratios vertically and horizontally. In contrast to *dimain*, *belian* ‘something bought’

[1]In (Chan, 2008, p. 79), saturation is the maximal proportion of word forms attested for any one lemma of a given paradigm. Here we use the term for each entire grid.

Table 2.2: Statistics on the Bible corpus for the four languages.

Language	**# of tokens (N)**	**# of types (V)**	**Length of types avg±std. dev.**
English	792,074	12,498	7.03 ± 2.18
Indonesian	648,606	15,641	7.84 ± 2.63
Modern Greek	706,771	36,786	8.49 ± 2.49
Russian	560,524	47,226	8.26 ± 2.73

is a valid Indonesian word although it does not appear in the example text in Figure 2.1. The issues on filling empty cells inside analogical grids are addressed in Section 2.5.

2.4.2 The Bible corpus

We carried out experiments on a multilingual parallel corpus created from the translation of the Bible, both the Old and New Testaments, collected by Christodoulopoulos[1]. This corpus is a continuation of previous efforts described in (Resnik et al., 1999). We selected four languages with different richness in morphology: English, Russian, Modern Greek, and Indonesian. The reason for using a multilingual parallel corpus is the need to draw conclusions across different languages in a reliable way. Table 2.2 presents statistics on the corpus. For each text in each language, we first extracted the list of all words, then produced all analogical clusters, and finally built all analogical grids.

2.4.3 Analogical grids produced from the Bible corpus

Table 2.3 shows the number of analogical grids produced in each language. These numbers show that English produced the lowest number of analogical grids. Indonesian produced twice as many tables as English. Modern Greek and Russian produced five times more tables than English. Modern Greek produced a larger amount of analogical grids than Russian despite its lesser number of analogical clusters. To summarize, languages with poorer morphology tend to produce less analogical grids than languages with richer morphology, which meets intuition. Figure 2.5 plots the number of analogical clusters and analogical grids obtained for each language against their size.

[1] http://homepages.inf.ed.ac.uk/s0787820/bible/

Table 2.3: The number of analogical clusters and number of analogical grids produced from the Bible corpus in each language with the time needed to produce them

Language	**# of clusters**	**# of grids**	**Total time (h:min)**
English	593,129	12,855	0:45
Indonesian	1,491,415	25,752	2:04
Modern Greek	4,068,913	69,173	11:03
Russian	4,762,509	60,035	10:34

Let us recall that, by construction, contrary to many previous works in morphological induction (Schone and Jurafsky, 2000; Goldsmith, 2001; Dryer and Eisner, 2011), our analogical grids do not contain in any way information about word frequency, word context, nor the frequency or distribution of morphemes or the like. The extraction method is agnostic from two points of view. Firstly, no semantic information is present during the extraction process. Secondly, it organises word forms in analogical grids by relying on well-defined formal relationships between words (ratios) and is thus language-independent[1]. The extracted relations put words into a series of equal ratios before organizing them into analogical grids. The relations hold on the level of characters and are thus purely formal. This means that the method does not make any a priori linguistic assumption, and just operates at the level of characters. Its application to different languages is thus possible and can lead to cross-linguistic comparisons on the respective complexity of the processed languages.

[1]However, it is not independent of the properties of the writing system used.

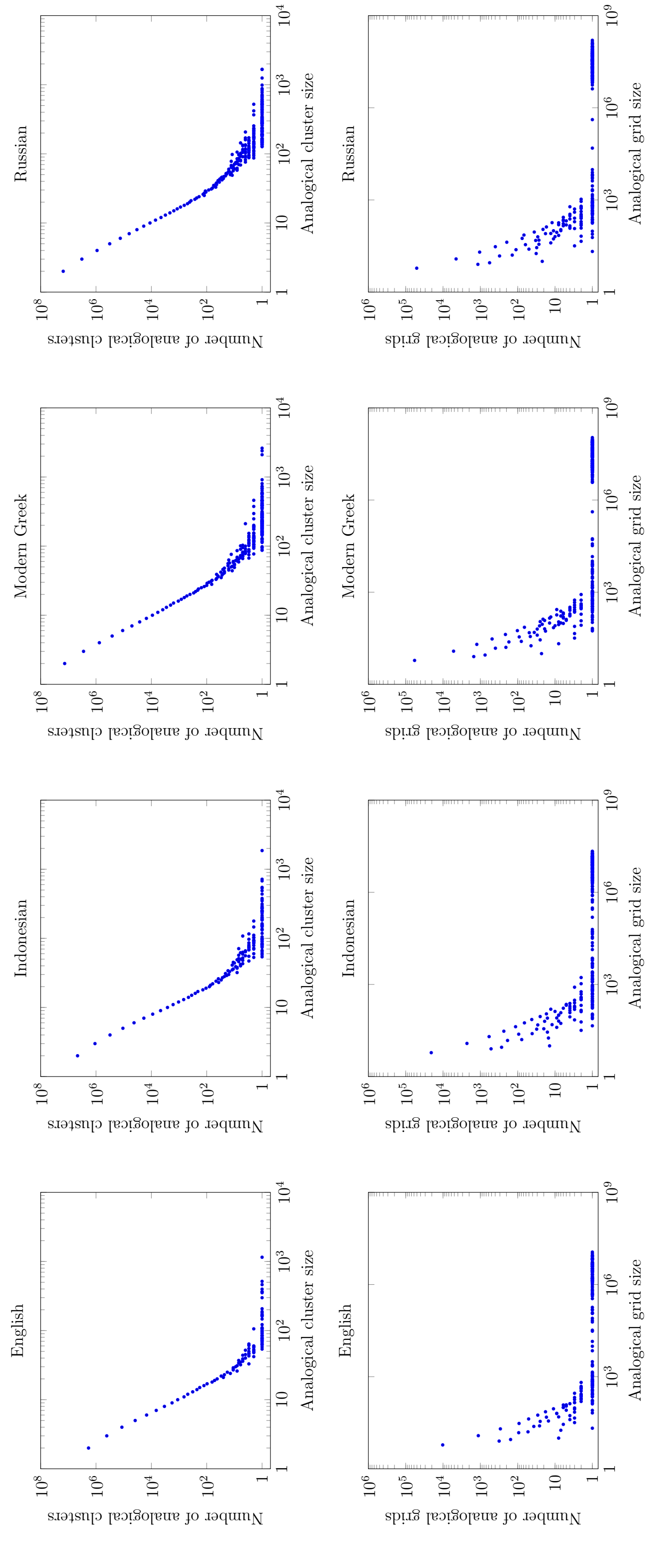


Figure 2.5: Number of analogical clusters with the same size in each language *(top)* and number of analogical grids with the same size in each language *(bottom)*. Logarithmic scale on both axes. From *left* to *right*: English, Indonesian, Modern Greek and Russian. Same ranges along the axes for all languages.

2.4.4 Analysis of the size and saturation of analogical gridsacross languages

The graphs at the top of Figure 2.6 show the number of analogical grids with the same sizes in each language. Most of the analogical grids have a small size. The number of analogical grids with the same size decreases gradually as the size increases. The plots for languages with a richer morphology are naturally shifted to the upper right of the graph in comparison with the poorer ones. The interpretation is that languages with a richer morphology produce bigger analogical grids on average and also more analogical grids for a given size. All of this meets intuition.

We now turn to the study of the saturation of analogical grids compared to their size. The top of Figure 2.6 shows saturation against size for analogical grids in each language. Analogical grids with smaller sizes tend to have higher saturation. As the smallest analogical clusters have three word pairs, the minimal size for an analogical grid is 6. In each graph, the top left point stands for analogical grids with this minimal size of 6; by construction, they have a saturation of 100 %. Some tables are extremely sparse. Because of the logarithmic scale on the y-axis, the bottom half is for tables with a saturation of less than 1 %.

In all cases, the plots exhibit a similar linear shape in logarithmic scale across all languages. This would correspond to Formula (2.7). We confirmed the similarity by the computation of the coefficients a and b for each language, as obtained by the least squares method. These coefficients are presented in Table 2.4. They are almost the same in all languages. This confirms the view given in Figure 2.6 that they superimpose almost perfectly.

$$\log(\text{saturation}) = a \times \log(\text{size}) + b \tag{2.7}$$

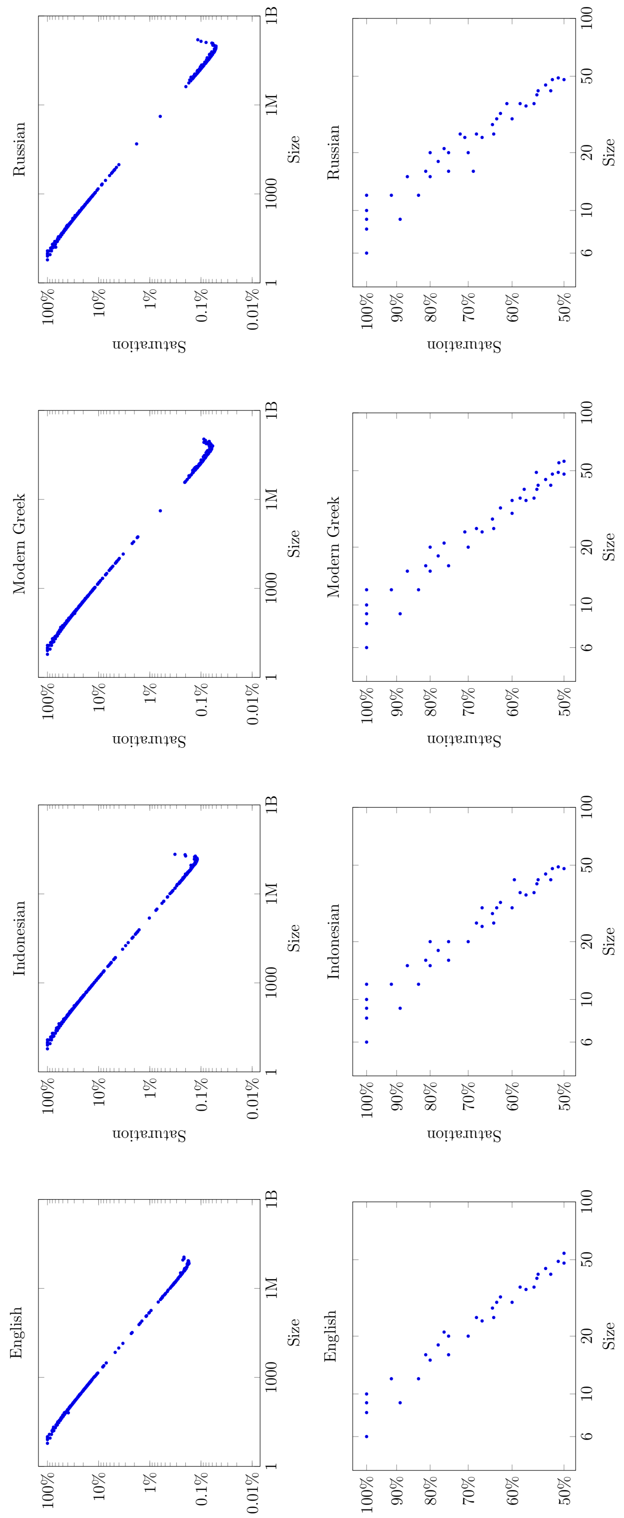


Figure 2.6: Saturation of analogical grids against size in each language. From *left* to *right*: English, Indonesian, Modern Greek and Russian. Algorithmic scale on both horizontal (size) and vertical (saturation) axes. Saturation (in ordinates) in the range [0 %, 100 %] *(top)* and in the range [50 %, 100 %] *(bottom)*. Same ranges along the horizontal axes for all languages for the same range of saturation.

Table 2.4: Linear coefficients for each language; and for different sizes and different genres in English.

Language	**Data and size**	**Range for saturation**			
		[0%,100 %]		**[50%,100 %]**	
		a	***b***	***a***	***b***
English	Bible 100.0 %	-0.480	0.510	-0.366	0.332
	50.0 %	-0.479	0.507	-0.372	0.343
	25.0 %	-0.476	0.499	-0.368	0.336
	12.5 %	-0.474	0.491	-0.361	0.323
	Europarl = Bible	-0.481	0.516	-0.365	0.333
Indonesian	Bible 100.0 %	-0.481	0.518	-0.371	0.343
Modern Greek	”	-0.479	0.514	-0.369	0.342
Russian	”	-0.482	0.520	-0.370	0.342

Let us make a first remark on the type of the observed relation. This is not yet another instance of a Zipfian law, because, in the present case, the objects are not ranked individually according to their frequency (number of occurrences). In Zipfian law, the x-axis stands for the list of individual objects ranked by frequency. Recall also that our analogical grids do not encapsulate any information about the frequency of individual words whatsoever. In our graphs, two analogical grids with the same size have the same abscissa. If they also have the same saturation, they have the same ordinate and are thus plotted as the same point. To make it clear, we plot the frequency of analogical grids as the third axis for English in Figure 2.7.

The interesting fact that comes into light is not so much the fact that the relation between size and saturation of analogical grids is a log–log relation, but the fact that it exhibits very similar slopes in all four languages. A reasonable explanation is that these coefficients are independent of the language because they characterize the corpus used. The corpus is defined by its size and its genre.

We first inquired whether the coefficients depend on the size of the corpus used. We performed the same experiment in English and let the size of the corpus vary: a half, a quarter, and an eighth of the original size. The computation of the coefficients led to very similar results as shown in Table 2.4.

We then inquired about the influence of the genre and performed the same experiment with the same size of text in English again. We chose the

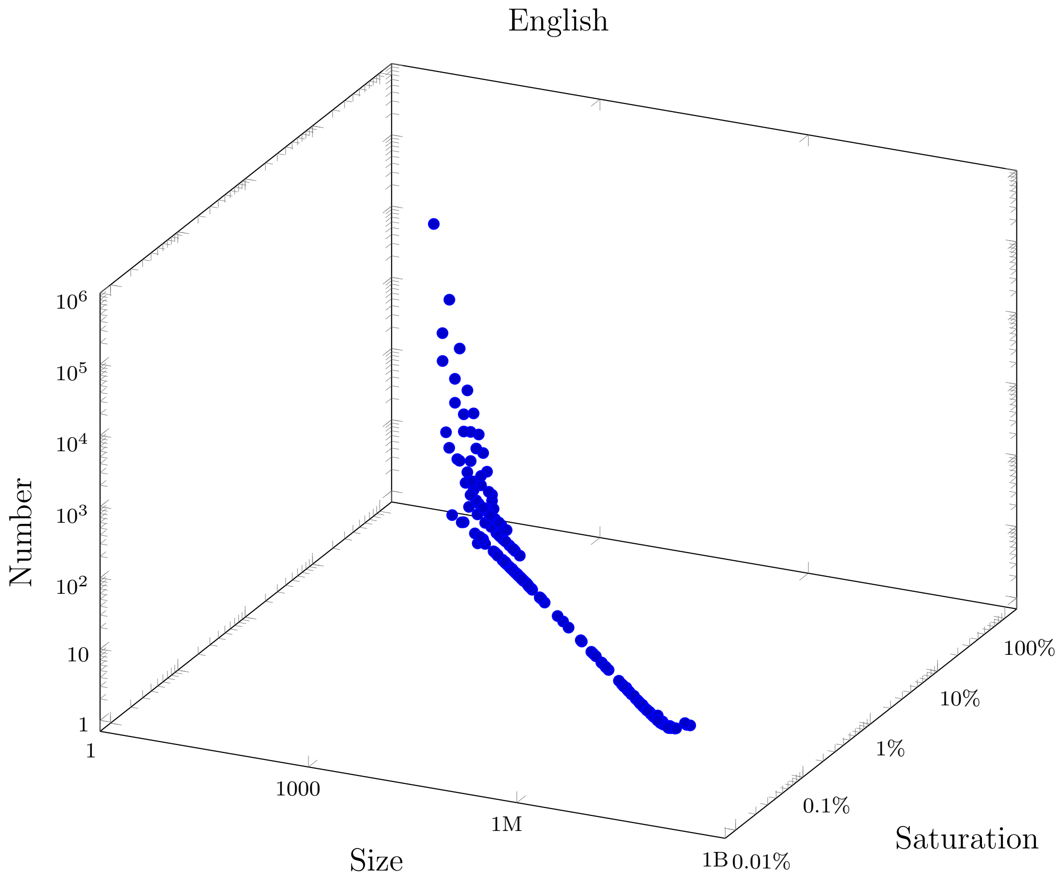


Figure 2.7: Number of analogical grids obtained against their size and saturation in English. Algorithmic scale on the three axes.

Europarl corpus for this experiment. Again, the computation of the linear coefficients led to very similar results, as shown in Table 2.4. Further experiments with more parameters varying are obviously required to confirm this. As for the time being, we conclude that we have found a relatively stable phenomenon concerning paradigm tables, across languages with different richness in morphology.

Rather than in the type of relation and the slope, the differences between languages should thus be looked for in the differences that can be observed in the middle and the tail of each graph, and in the difference in the number of analogical grids each point stands for in each graph (not visible in Figure 2.6 as mentioned at the beginning of this section, but visible in the bottom graphs of Figure 2.5).

2.5 Study on filling empty cells in analogical grids

In this section, we address the problem of filling empty cells in analogical grids and checking for the validity of the words produced. Algorithms for this task have been proposed (Lepage, 1998; Yvon, 2003; Langlais and Patry, 2007), as transducers and modified versions of Levenshtein automata (Schulz and Mihov, 2002) can be designed from Formula (2.1) to output words that fill in blank cells.[1] Here, we choose to carry out the experiments in Indonesian, a language known for its richness in derivational morphology where word forms often change their category (derived) when affixing is performed. The confidence of filling an empty cell can be measured by using a statistical test, for example, Fisher's exact test.

We would also like to check for the validity of the word forms newly generated by filling empty cells in analogical grids. Checking for the validity of the word produced can be done by relying, for example, on information theoretical considerations (Goldsmith, 2001), semantic features acquired by techniques like LSA (Dryer and Eisner, 2011), in addition to parts of speech, or the use of word embeddings (Soricut and Och, 2015). In this work, we consider using:

- **morphological analyser** for the level of morphology, and
- **distributional semantic representation**: for the level of semantic

[1]Filling one million of cells by solving one analogy for each cell takes 1 second.

Form	makan : makanan :: minum : minuman
Morphology	makan_VB : makan+an_NN :: minum_VB : minum+an_NN
Semantic	$\vec{\text{makanan}} - \vec{\text{makan}} + \vec{\text{minum}} \approx \vec{\text{minuman}}$

Figure 2.8: Confirming an analogy on different levels of representation: form, morphology, and semantic for the word *minuman*.

Figure 2.8 illustrates how we confirm the explanation of an Indonesian word form, *minuman*, on the three levels of surface form, morphology, and distributional semantic at the same time.

2.5.1 Validating the explanation of unseen words

We carried out a ten-fold cross-validation experiment using the BPPT[1] corpus provided by PAN Localization[2]. BPPT is an Indonesian-English aligned parallel corpus of news articles. The Indonesian part contains almost half a million tokens (words in the corpus) representing twenty-seven thousand types (number of different words). The average length of a token is around six characters while the average length for types is almost eight characters. Almost half of the tokens (44.3%) are hapaxes. Each of the ten test sets contains around 1,200 unseen words (almost 15% of the test set). The statistics for the data, training and test sets, are shown in Table 2.5.

The experimental results show that, on the level of form only, 97% of the unseen words can always be explained. A manual inspection of the data showed that the remaining unseen words are proper nouns and marked plurals. Around 80% of the unseen words explained on the level of form can also be explained on the level of morphological representation. More than 55% of the unseen words explained on the level of form can also be explained on the level of semantic representation. Overall, 49% of the unseen words explained on the level of form could be explained on these two additional representation levels.

Examples of unseen words that can be explained or not on each level of representation taken from the first batch of ten-fold cross-validation are given in Table 2.6. The first row stands for unseen words that can be explained on the level of surface form but not on the other two levels. These words are nouns and proper nouns although there are also a few numbers of inflected forms. The unseen words that can be explained on both surface

[1]Licence: Creative Commons BY-NC-SA 3.0
[2]`http://www.panl10n.net/indonesia/`

Table 2.5: Number of types in training and test set for each experiment batch (*left*). Number of unseen words (*right*) explained on the level of: form (F), morphological representation (M); semantic representation (S).

exp	**# types**		**# unseen words**				
				explained			
	training	**test**	**total**	**F**	**F∩M**	**F∩S**	**F∩M∩S**
1	26,039	8,629	1,276	1,249	1,010	787	721
2	26,110	8,533	1,205	1,186	946	612	540
3	26,030	8,654	1,285	1,255	1,017	685	625
4	26,029	8,732	1,286	1,262	1,031	712	637
5	26,063	8,832	1,252	1,234	1,012	674	599
6	26,163	8,532	1,152	1,131	910	587	536
7	25,948	8,823	1,367	1,343	1,098	791	712
8	26,020	8,712	1,295	1,269	1,031	673	616
9	26,089	8,646	1,226	1,207	992	664	603
10	26,025	8,667	1,290	1,268	1,000	662	587

Table 2.6: Examples of unseen words explained or not on each level of representation: surface form (F), morphological representation (M), and distributional semantic representation (S).

F	**M**	**S**	**Number**	**Examples**	**English translation**
✓	×	×	172	*ilustrasi* *terenggut* *Montolivo*	‘illustration‘ ‘wrenched‘ person’s name
✓	✓	×	286	*disewakan* *bercampur* *menyepakatinya*	‘for rent‘ ‘mixed‘ ‘to agree‘
✓	×	✓	67	*endoplasma* *perfeksionis* *radjawali*	‘endoplasm‘ ‘perfectionist‘ name of a kind of bird
✓	✓	✓	724	*persilangan* *terkoordinasi* *pembelajaran*	‘crossing‘ ‘coordinated‘ ‘learning‘

form and morphological representation levels but not on the distributional semantic representation level are mostly inflected forms. These words can be generated by adding prefix or suffix to a lemma or other inflected form (morphological phenomena that are captured by our formalisation) which meets our expectation. The last row shows the unseen words that can be explained on the three different levels at the same time. These words are also mostly inflected forms of nouns and verbs. For example, *persilangan*, *terkoordinasi*, and *pembelajaran* are inflected forms of *silang*, *koordinasi*, and *belajar*. Nouns are the dominant category for unseen words that can be explained on the level of surface form and distributional semantics but not on the morphological representation level.

2.5.2 Confidence of filling empty cells

We perform experiments in measuring the confidence of filling empty cells in analogical grids on `idn-tagged-corpus`. This corpus is a POS-tagged corpus manually annotated by native speakers of Indonesian. It is based on the BPPT corpus, which is used in Section 2.5.1.

We extract all the analogical grids from the list of word forms contained in the first thousand lines of `idn-tagged-corpus`. We use two different word vector representations, characters-only and characters + POS feature vectors. For each word vector representation, we construct the analogical grids while maintaining a saturation threshold. We choose to use 50% and 90% as our saturation threshold when building the analogical grids with the intuition that empty cells in grids with higher saturation will be more reliable to fill. We then use Fisher's exact test to measure the confidence of filling an empty cell.

Fisher's exact test is a statistical test to analyse a contingency table. Fisher (1922) showed that the hypergeometric distribution of the numbers in the tables can be used to calculate the significance of the observation from a null hypothesis. It is usually used for 2 x 2 contingency tables, but it is not limited to them. Pedersen (1996) reported that Fisher's exact test is a more appropriate test to identify dependent word pairs in comparison to other statistical methods. Here, we use Fisher's exact test to measure the confidence of filling an empty cell. Before filling an empty cell P_i^j, we create a 2 x 2 table by observing the row$_i$ and column$_j$. The p-value p is calculated as follows.

	row$_i$	column$_j$
# non-empty cells	a	b
# empty cells	c	d

$$p = \frac{(a+b)! \quad (c+d)! \quad (a+c)! \quad (b+d)!}{a! \quad b! \quad c! \quad d! \quad (a+b+c+d)!} \tag{2.8}$$

Table 2.7 shows the number of newly generated word forms obtained from filling analogical grids built under different configurations. From the point of view of feature vectors, we can see that using the characters-only feature vectors will give us analogical grids with more empty cells. Thus, we generate more new word forms but face the drawbacks of producing more invalid word forms. On the contrary, characters + POS feature vectors deliver a smaller number of newly generated word forms. However, they are around twice as good in terms of the ratio of valid generated word forms.

As can be seen from Table 2.7, the use of Fisher's exact test under the condition of p-value $\leq 5\%$ gives 29% (= 24/82) and 65% (= 11/17) ratio of validated generated word forms. The p-value from Fisher's exact test leads to being very cautious in filling the empty cells. A consequence of that is that no empty cell is filled under a saturation threshold of 90%. However, it is around two times better performance in comparison to the configuration without Fisher's exact test, 15% (= 2,426/15,886) and 38% (= 904/2,401).

Table 2.7: Number of newly generated word forms with and without Fisher's exact test. The validity of the newly generated word forms is checked against a rule-based morphological analyser.

Feature vector	Saturation threshold (%)	# empty cells	# generated word forms			# valid generated word forms		
			w/o Fisher	w/ Fisher		w/o Fisher	w/ Fisher	
				$p \leq 5\%$	$p > 5\%$		$p \leq 5\%$	$p > 5\%$
char	≥ 50	34,914	*15,886	**82**	15,824	*2,426	**24**	2,409
	≥ 90	140	93	0	92	23	0	23
char + POS	≥ 50	4,313	*2,401	**17**	2,387	*904	**11**	897
	≥ 90	19	16	0	15	7	0	7

2.6 Summary of the chapter

We proposed a mathematically well-defined data structure called analogical grids. We introduced a novel method to automatically extract analogical grids from words contained in a corpus. Analogical grids can be seen as a step towards the automatic production of paradigm tables which are usually produced manually by grammarians or linguists relying on grammatical tradition and by thorough linguistic formalisation.

We observed an interesting phenomenon when producing analogical grids in four different languages on translations of the same text. It relates the saturation of the obtained analogical grids to their size. Experimental results show that the coefficients which characterize the relation would not be influenced by the size, the genre or the language of texts. This brings us to lay the hypothesis that this particular phenomenon might hold in any language.

We carried out experiments in Indonesian, a language known for its rich derivational morphology, to confirm the explanation of the unseen words. We first explain the unseen words on the level of surface form. The explanations are then confirmed on two additional levels of representation: morphological and distributional semantic representation. Results from ten-fold cross-validation show that more than 98% of the unseen words can be explained on the level of surface form. The remaining unseen words are mostly: plurals (formed by repetition in Indonesian, which is excluded from our formalisation) and proper nouns. As a final result, almost half of the unseen words can be explained on three different levels: surface form, morphology, and distributional semantics at the same time.

Chapter 3

Morphological generation using analogical grids

The previous chapter described how to explain unseen words contained in a test set by using words contained in the training set. In this chapter, we address the issue of generating unseen words, particularly for the purpose of the inflection task. By discovering the relations to other word forms in a training data, we predict inflected word forms by solving analogical equations.

3.1 Organisation of the chapter

This chapter is organised as follows: Section 3.3 presents the background of the morphological inflection task and how the concept of analogical grids can be used to tackle the task. Section 3.4 introduces basic notions related to analogical grids. Section 3.5 provides an overview of the data used to carry out the experiments. Section 3.6 explains how to perform data augmentation and generate more data using standard transducer automata. Section 3.7 presents our experiments in more than 100 languages with different richness in morphology. Section 3.8 analyses the results and explores the relationship between the saturation and the size of analogical grids and the improvement in results. Section 3.9 presents further discussion and analysis of the experimental results. Section 3.10 gives a summary of the chapter.

3.2 List of publications

The research described in this chapter has been published in the following publications[1].

Journal paper

(J1) Fam, R. and Lepage, Y. (2022). Organising lexica into analogical grids: A study of a holistic approach for morphological generation under various sizes of data in various languages. *Journal of Experimental & Theoretical Artificial Intelligence*, 0(0):1–26

Conference paper with reviewing committee

(C7) Fam, R. and Lepage, Y. (2018a). IPS-WASEDA system at CoNLL–SIGMORPHON 2018 shared task on morphological inflection. In *Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection (CoNLL–18)*, pages 33–42, Brussels. Association for Computational Linguistics

(C10) Fam, R. and Lepage, Y. (2017a). A holistic approach at a morphological inflection task. In *Proceedings of the 8th Language and Technology Conference (LTC–2017)*, pages 88–92, Poznań, Poland. Fundacja uniwersytetu im. Adama Mickiewicza

[1]Numbering follows the document *04-Research achievements publications* submitted together for the degree application.

input		output	
Lemma:	*to illustrate*		
Target MSD:	`Category = verb`		
	`Person = 3`	**Target form**:	*illustrates*
	`Number = singular`		
	`Tense = present`		

Figure 3.1: An example of morphological inflection task in English: given the lemma *to illustrate* and the morphosyntactic description of the target form (target MSD), generate the form *illustrates*.

input		output	
Lemma:	*illustrate*	**Target form**:	*illustrates*
Target MSD:	`V;3;SG;PRS`		

Figure 3.2: Question shown in Figure 3.1 written as SIGMORPHON data. The MSFs are written according to Unimorph schema. See Section 3.5 for further explanation.

3.3 Introduction and background

In this section, we describe the morphological generation task, particularly the reinflection task. We introduce the application of the novel concept of analogical grids as a holistic approach to the morphological generation task.

3.3.1 Morphological inflection task

We address the problem of morphological inflection task:

> given a **lemma** (e.g. the dictionary form of a word) and the target form's **morphosyntactic description**, generate the **target form**, i.e., an inflected form of the lemma.

A morphosyntactic description (MSD) consists of morphosyntactic features (MSF) which describe the target form. These features are usually found in the output of morphological analyzers. MSDs may differ between languages depending on how much morphological complexity the languages exhibit. This subject will be addressed in the discussion in Section 3.9.3. Languages with more complex morphology will have a larger number of MSFs. Figure 3.1 shows an example of completing the morphological inflection task: generating the third singular present form of the lemma *to illustrate* in English. Figure 3.2 shows how the task is described with SIGMORPHON data.

This task has been promoted heavily in recent years by the Association for Computational Linguistics (ACL) Special Interest Group on Association

talk :	*talks* :	*talking* :	*talked*
love :	*loves* :	:	
like :	*likes* :	:	
walk :	:	*walking* :	
read :	*reads* :	*reading* :	

fast :	*faster* :	*fastest* :	*fastly*
smooth :	*smoother* :	*smoothest* :	*smoothly*
hard :	:	:	*hardly*

Figure 3.3: Analogical grids in English.

talk :	***talks*** :	*talking* :	*talked*
love :	*loves* :	:	
like :	*likes* :	:	
walk :	:	*walking* :	
read :	*reads* :	*reading* :	
illustrate :	***x*** :	:	

$$\textit{talk} : \textit{talks} :: \textit{illustrate} : x$$
$$\Rightarrow$$
$$x = \textbf{\textit{illustrates}}$$

Figure 3.4: Generating a target form using analogical grid

for Computational Morphology and Phonology (SIGMORPHON) with its evaluation campaign[1]. Also, in its 2020 edition, the International Colloquium on Grammatical Inference (ICGI 2020) promoted the same task 'with some modifications and a focus on diversity in languages.'[2]

3.3.2 Analogical grids in morphological generation

Figure 3.3 shows two examples of analogical grids, in English. Analogical grids can be used to generate target forms by exploiting the relation between forms which is captured in analogical grids, the relation of formal analogy. Analogy is a relation between four objects: A, B, C, and D where A is to B as C is to D. Target forms can be coined by solving analogical equations using words from analogical grids. Figure 3.4 demonstrates how to generate the target form *illustrates* by taking a pair from the first two columns of the analogical grid previously shown in the left part of Figure 3.3. In this case, we state that: *talk* is to *talks* as *illustrate* is to *illustrates*.

In this chapter, we investigate the advantage of organising a lexicon as a set of analogical grids to improve performance in the morphological inflection task. We carry out experiments on the SIGMORPHON dataset which is used in the Morphological Reinflection Shared Task. The experimental results show that our holistic approach always performs better than the morpheme-based approach on all different sizes of training datasets. We also observe that the use of data augmentation helps improve the performance of the neural approach performance in low-resource conditions. However, there is

[1] `sigmorphon.github.io/sharedtasks/`
[2] `aryamccarthy.github.io/icgi2020/` The quote is from this page.

a trade-off between performance and time to train the system. We also find that data augmentation might not improve the performance any more after some point.

3.3.3 Contributions

The contributions of this chapter are summarised as follows:

- We introduce a holistic approach based on the notion of analogical grids to organise lexica and apply it to morphological inflection task;
- We investigate the comparison between performances of morpheme-based, holistic, and neural approaches in more than 100 languages with various morphological richness;
- We investigate the improvement of performance according to the size of the data; and
- Based on previously mentioned results, we analyse the influence of the granularity of units (morpheme or whole-word) and morphological complexity of the language on the systems' performance.

3.4 Basic notions

In this section, we present again the basic notions related to analogical grids and how to leverage them for the morphological task.

3.4.1 Illustration with toy data

In the sequel of this introduction, we illustrate each notion with an example. Figure 3.5 shows some samples of the SIGMORPHON data for English. From the list of lemmas and target forms contained in Figure 3.5, we can extract the analogical grid shown in Figure 3.6. However, we can immediately observe that nouns and verbs are being mixed inside the same analogical grid. This issue will be addressed in Section 3.4.4.

In standard linguistics, a systematisation of the relations between word forms is given by paradigm tables, which is the result of linguistic formalisation. Paradigm tables usually can be found in dictionaries (Figure 3.7). The difference between analogical grids and paradigm tables is that there are no exponents (*infinitive, preterit, etc.*) in analogical grids as is found in paradigm tables. Paradigm tables are products of linguistic studies and are

Lemma	Target form	Target MSD
illustrate	*illustrate*	`V;NFIN`
illustrate	*illustrates*	`V;3;SG;PRS`
illustrate	*illustrated*	`V;PTCP;PST`
create	*create*	`V;NFIN`
create	*creates*	`V;3;SG;PRS`
create	*creating*	`V;PTCP;PRS`
fuse	*fuse*	`V;NFIN`
fuse	*fused*	`V;PTCP;PST`
fuse	*fusing*	`V;PTCP;PRS`
illustration	*illustration*	`N;SG`
illustrate	*illustrations*	`N;PL`
creation	*creation*	`N;SG`
creation	*creations*	`N;PL`
see	*see*	`V;NFIN`
seed	*seed*	`N;SG`
seed	*seeds*	`N;PL`
go	*gone*	`V;PTCP;PST`
go	*goes*	`V;3;SG;PRS`

MSF	Feature	Value
`V`	Part-of-speech	Verb
`N`	Part-of-speech	Noun
`PTCP`	Part-of-speech	Participle
`NFIN`	Finiteness	Non-finite
`3`	Person	Third
`SG`	Number	Singular
`PL`	Number	Plural
`PRS`	Tense	Present
`PST`	Tense	Past

Figure 3.5: English SIGMORPHON data contain only verbs. For the purpose of our example, we added some noun descriptions.

illustrate	:	*illustrates*	:	*illustrated*	:	
create	:	*creates*	:		:	*creating*
fuse	:		:	*fused*	:	*fusing*
illustration	:	*illustrations*	:		:	
creation	:	*creations*	:		:	
see	:		:	*seed*	:	

Figure 3.6: An analogical grid created from the word forms given in Figure 3.5

	Infinitive	*Preterit*	*Past participle*	*Present participle*
Regular verb	walk smoke	walked smoked	walked smoked	walking smoking
Irregular verb	write think	wrote thought	written thought	writing thinking

Figure 3.7: A paradigm table taken from a French & English dictionary (Mansion, 1981)

$$\begin{gathered} \textit{illustrate} : \textit{illustrates} :: \textit{create} : \textit{creates} \\ \textit{create} : \textit{fuse} :: \textit{creating} : \textit{fusing} \\ \textit{fused} : \textit{illustrated} :: \textit{fuse} : \textit{illustrate} \end{gathered}$$

Figure 3.8: Some analogies extracted from analogical grid in Figure 3.6.

created manually. Here, we agnostically extract analogical grids relying on a formal relationship between words, called analogy.

3.4.2 Analogical grids

Let us remember again the notion of analogical grids (See Section 2.3.3). An analogical grid is a table of dimension $M \times N$ as defined by Formula (2.4). As illustrated by Figure 3.3, analogical grids extracted from texts usually contain empty cells. Another example is Figure 3.6. It is an analogical grid created from the set of English words contained in Figure 3.5.

According to Formula (2.4), we can get many analogies from the analogical grids of Figure 3.6. Figure 3.8 shows three of them. For example, the first analogy *illustrate* : *illustrates* :: *create* : *creates* states that *illustrate* is to *illustrates* as *create* is to *creates*.

3.4.3 Word ratios: Formal level

Each word is represented using the vector shown in Formula (3.1). The vector of the number of occurrences of each character in a string is called the Parikh vector of the string. The number of dimensions of the vector is the size of the alphabet. See again Section 2.3.1.

$$A \triangleq \begin{pmatrix} |A|_a \\ |A|_b \\ \vdots \\ |A|_s \\ \vdots \\ |A|_z \end{pmatrix} \qquad illustrate = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \tag{3.1}$$

The ratio between two words A and B is defined as the difference between the feature vectors of A and B. Formula (3.2) gives the definition of the ratio between the word *illustrate* and *illustrates*.

$$A : B \triangleq \begin{pmatrix} |A|_a - |B|_a \\ |A|_b - |B|_b \\ \vdots \\ |A|_s - |B|_s \\ \vdots \\ |A|_z - |B|_z \\ d(A, B) \end{pmatrix} \qquad illustrate : illustrates = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ -1 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{3.2}$$

3.4.4 Towards paradigm tables: MSD as feature

By using MSDs as features, instead of characters, in the dimensions of vector representation for words, we can extract paradigm tables from the words contained in datasets, like the SIGMORPHON dataset. In contrast with the use of Parikh vectors, which focus on the level of surface form of the words, here we take into account the morphological description of the words. Thus, we may have irregular forms inside the same analogical grid along with regular forms as long as they are described with the same MSD. This is simply done by embedding MSFs as boolean features, as shown in Formula (3.3). Here, $|S|_{isT}$ stands for whether the feature T is present for word form S.

$$A : B \triangleq \begin{pmatrix} A_{isLEMMA} - B_{isLEMMA} \\ A_{isVERB} - B_{isVERB} \\ \vdots \\ A_{isPRESENT} - B_{isPRESENT} \end{pmatrix} \qquad illustrate : illustrates = \begin{pmatrix} 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} \tag{3.3}$$

Figure 3.9 shows the analogical grids extracted using MSDs as features instead of characters and edit distance. We obtain two analogical grids which

$$\begin{array}{ccccccc} \textit{illustrate} & : & \textit{illustrates} & : & \textit{illustrated} & : & \\ \textit{create} & : & \textit{creates} & : & & : & \textit{creating} \\ \textit{fuse} & : & & : & \textit{fused} & : & \textit{fusing} \\ \textit{go} & : & \textit{goes} & : & \textit{gone} & : & \end{array} \qquad \begin{array}{ccc} \textit{illustration} & : & \textit{illustrations} \\ \textit{creation} & : & \textit{creations} \\ \textit{seed} & : & \textit{seeds} \end{array}$$

Figure 3.9: Analogical grids extracted by using MSD as features. The verbs (*left*) and nouns (*right*) are separated into two separate analogical grids. Also notice that now irregular forms may enter the analogical grid.

separate the verbs and the nouns. We immediately notice that now irregular forms can enter the analogical grid, in this case, the last line of the analogical grid on the *left*: (*go* : *goes* : *gone*).

3.4.5 Level of form and morphological at the same time

An astute reader will notice immediately that the use of characters and edit distance only might allow a mixture of actual conjugation and mere coincidence in the data. For example, the verb pair (*illustrate*, *illustrates*) has the same ratio as the noun pair (*illustration*, *illustrations*). Both pairs share the same suffix '∼s' but are described with different MSDs, primarily the part of speech tag: noun and verb. The same thing goes with the use of MSD. It mixes up the regular and irregular forms inside the same analogical grid, which may cause problems when generating a target form.

$$A : B \triangleq \begin{pmatrix} |A|_a - |B|_a \\ |A|_b - |B|_b \\ \vdots \\ |A|_s - |B|_s \\ \vdots \\ |A|_z - |B|_z \\ d(A,B) \\ A_{isLEMMA} - B_{isLEMMA} \\ A_{isVERB} - B_{isVERB} \\ \vdots \\ A_{isPRESENT} - B_{isPRESENT} \end{pmatrix} \qquad \textit{illustrate} : \textit{illustrates} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ -1 \\ \vdots \\ 0 \\ 1 \\ 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} \tag{3.4}$$

To avoid this, we combine the character counts and edit distance with MSD information as features when computing the ratio between two words (see Formula (3.4)). In doing so, the verb pair (*illustrate*, *illustrates*) is sep-

illustrate	:	*illustrates*	:	*illustrated*	:			*illustration* :	*illustrations*
create	:	*creates*	:		:	*creating*		*creation* :	*creations*
fuse	:		:	*fused*	:	*fusing*		*seed* :	*seeds*

Figure 3.10: Analogical grids extracted by taking into account: the level of form (characters and edit distance) and the level of morphological (MSD). The verbs (*left*) and nouns (*right*) are separated into two separate analogical grids. Furthermore, there is no more mixture of regular of irregular forms inside the analogical grids due to stricter constraints. The grid on the right contains only 2 columns which is a degenerated form of analogical grid, called analogical cluster.

arated into a different grid from the noun pair (*illustration*, *illustrations*). In this way, not only the formal aspect of the transformation taking place on the formal level (characters and edit distance) but also on the morphological level (between MSDs) is taken into account and encapsulated in the analogical grids. Using the above definition, we are able to separate the two grids shown in Figure 3.10.

3.5 Languages and data

We carry out our experiments on SIGMORPHON 2018 Shared Task: Morphological Reinflection Task dataset. This dataset was developed specifically for the inflection task. It contains data from 103 different languages.

Basically, the languages are mainly from the Indo-European family, with members of the Indo-Aryan, Iranian, Germanic, Slavic or Romance subfamilies, with, geographically speaking, a strong representation of languages from Europe like German, Livonian, Norwegian, Occitan, Sorbian, etc., some of them in their modern or old forms, like Latin, Old-French, Middle-French, Norman and modern French; but other languages from other regions of the world are also to be found like Arabic, Quechua, classical Syriac, Murrinh-Patha, Navajo, Tibetan or Swahili.

3.5.1 Data format and size

The dataset consists of lines of triplets. A triplet consists of a lemma, a target form, and a target MSD separated by tabulation characters. A target MSD consists of several MSFs separated by semicolons ';'. The MSFs are coded according to the Unimorph Schema (Kirov et al., 2018). For example, `V` stands for *verb*, while `PRS` stands for *present*.

The provided resources are categorised into:

- **train**: this dataset is the dataset which can be manipulated by the participants to solve the task. It comes in three different sizes: low, medium, and high. Most of the languages have all of the three sizes. Sixteen have only *low* and *medium* training datasets and no *high* training dataset. One language has only the *low* training dataset: Telugu. See Table below.

Data	Size	Exceptions (no data)
low	100	
medium	1,000	Telugu
high	10,000	Telugu, Cornish, Greenlandic, Inggrian, Karelian, Kashubian, Kazakh, Khakas, Mapudungun, Middle-Low-German, Middle-High-German, Murrinhpatha, Norman, Old-Irish, Scottish-Gaelic, Tibetan, Turkmen.

- **dev**: this dataset is used as a validation set during the training phase. It consists of 1,000 lines for most of the languages. Some languages have less than 1,000 lines.

Size	Language
50	Cornish, Greenlandic, Inggrian, Karelian, Kashubian, Kazakh, Khakas, Mapudungun, Middle-High-German, Middle-Low-German, Murrinhpatha, Norman, Old-Irish, Scottish-Gaelic, Telugu, Tibetan, Turkmen.
100	Azeri, Bengali, Breton, Classical-Syriac, Crimean-Tatar, Friulian, Haida, Kabardian, Kannada, Ladin, Livonian, Maltese, Neapolitan, North-Frisian, Occitan, Old-Church-Slavonic, Pashto, Swahili, Tatar, Uzbek, Votic, Welsh, West-Frisian, Yiddish .

- **test**: this dataset is to evaluate the performance of the system. It consists of 1,000 lines for most of the languages, with exceptions the same as the *dev* dataset.

Table 3.1: Statistics of the features found in the dataset given. Numbers for unseen features are against *train* dataset. Caution: number of rules and unseen rules are based on the rule extraction method explained in Section 3.6.

Feature	low			medium			high		
	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
Characters (train)	14	29	51	14	33	63	19	40	86
- unseen (dev)	0	4	21	0	1	8	0	0	4
- unseen (test)	0	5	24	0	2	10	0	0	3
Lemmata (train)	5	77	100	5	487	989	15	2,308	8,643
- unseen (dev)	0	414	984	0	295	960	0	98	743
- unseen (test)	0	415	985	0	295	957	0	97	764
MSFs (train)	5	22	43	5	23	48	7	25	48
- unseen (dev)	0	1	8	0	0	2	0	0	1
- unseen (test)	0	2	10	0	0	2	0	0	1
MSDs (train)	4	45	95	4	94	726	5	126	1,649
- unseen (dev)	0	44	695	0	8	414	0	0	6
- unseen (test)	0	44	682	0	8	402	0	0	8
Rules (train)	26	98	100	147	838	1,000	815	5,642	9,842
- unseen (dev)	12	561	997	30	504	995	22	398	971
- unseen (test)	19	562	1,000	32	503	996	20	395	969

Table 3.2: Statistics for unseen feature on *test* dataset relatively to *dev* dataset. Caution: number of rules and unseen rules are based on rule extraction method explained in Section 3.6.

Unseen feature	**Min**	**Avg**	**Max**
Characters	0	2	13
Lemmata	0	300	958
MSFs	0	0	4
MSDs	0	12	397
Rules	19	504	995

3.5.2 Statistics on the data

Let us now look at some statistics on the given dataset shown in Table 3.1. Overall, we observe a non-decreasing phenomenon from *low* to *high* for all of the number of pieces of information (features) found in the training dataset. On the opposite, we found a non-increasing pattern for the unseen information contained in the dev dataset relative to the training dataset which is shown in Table 3.2. This shows that bigger resources gradually cover the unseen data encountered in the smaller ones.

Norman, Telugu, Cornish, and Uzbek are languages with a smaller number of lemmata in the training dataset. However, these languages tend to have less, even zero for some languages, unseen lemmata relative to the dev dataset. They also have a smaller number of unseen characters. On the other hand, languages like Finnish, Russian, English, French, and German have the biggest number of unseen lemmata despite having the biggest number of lemmata in the training dataset compared to other languages.

Let us now turn to the number of MSFs and MSDs. These numbers can be interpreted as an estimation of how large or complex the paradigm for that particular language is. Basque, Quechua, Turkish, Zulu are languages with a higher variety of unique MSDs. Basque, in particular, has astonishingly more than 1,600 MSDs in comparison to the average of around 126 MSDs per language in *high* datasets. The same thing can be seen for *low* and *medium* data. Almost all of the lines are associated with different MSDs in the *low* training dataset. Furthermore, Basque also topped as the language with the highest number of unseen MSDs for all dataset sizes.

We also count the number of rules found in the dataset (see the last two rows in Table 3.1). These rules are not morphological rules defined by linguists but the ones extracted by the method explained in Section 3.6. For all languages and all datasets, we count how many unique rules can be

Table 3.3: Overview of number of productive and unproductive rules for all data sizes computed using Tolerance Principle (Formula (3.5)).

Data	**productive**			**unproductive**			**total**		
	Min	**Avg**	**Max**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**
low	1	20	34	0	25	90	4	45	95
medium	1	65	271	0	29	503	4	94	726
high	0	85	551	0	42	1,151	5	126	1,649

extracted and relatively unseen to the respective dev dataset. Telugu, Tatar, and Swahili are the languages with the lowest number of unseen rules. We expect to have good performance in these languages because it means that most of the transformations from the lemma into the target form are present in the training data. Figure 3.11 presents an excerpt of data in various languages with various writing systems.

3.5.3 Regularity and exceptions

Regulars and irregulars (or exceptions) are other phenomena that are interesting to observe. To estimate how many regulars and irregulars there are in the dataset, we consider the Tolerance Principle (Yang, 2016). It is a way to estimate whether a rule is productive or not based on its frequency in a given set of data. Let R be a rule observed over a set of N items; and e the number of items not supporting R. R is productive if and only if e does not exceed θ_N (see Formula (3.5)).

$$e \leq \theta_N = \frac{N}{\ln N} \tag{3.5}$$

We use the Tolerance Principle to observe the distribution of regular (productive) and irregular (unproductive) rules in the dataset. In our case, we define rule R as an MSD: (LEMMA, MSD). A generalised rule inside an analogical grid may be (LEMMA, MSD_1), (LEMMA, MSD_2), ... (LEMMA, MSD_n). In the analogical grid, it is represented by a line as follows (remember Figure 3.16).

$$\text{LEMMA} : \text{MSD}_1 : \text{MSD}_2 : \ldots : \text{MSD}_n.$$

Table 3.3 presents the number of productive and unproductive rules in our dataset. It can be observed that the number of both productive and unproductive rules rises from *low* data to *high* data. For both productive

Language	Lemma	Target form	Target MSD
Arabic	مَجَلَّةٌ	الْمَجَلَّتَيْنْ	N;DU;DEF;GEN
	مُسْتَشْفًى	مُسْتَشْفَيَاتِ	N;PL;PSSD;GEN
	قَالَ	يُقَلْنَ	V;3;PL;FEM;LGSPEC1;PASS
Armenian	բիբար	բիբարիցդ	N;ABL;SG;PSS2S
	մոռանալ	մոռացած	V;V.PTCP;RES
	հառաչախառն	հառաչախառններիդ	ADJ;DAT;PL;PSS2S
Belarusian	слова	словы	N;ACC;PL
	хварэць	хварэ́юць	V;PRS;3;PL
	чэшскі	чэ́шскія	ADJ;ACC;INAN;PL
English	*illustrate*	*illustrates*	V;3;SG;PRS
	exploit	*exploited*	V;V.PTCP;PST
	run	*running*	V;V.PTCP;PRS
Irish	*fótaidhé-óid*	*na fótaidhé-óidí*	N;NOM;PL;DEF
	comhaontaigh	*dá gcomhaontaíodh sibh*	V;2;PL;SBJV;PST
	uaigneach	*uaignigh*	ADJ;VOC+GEN;SG;MASV
French	*amoindrir*	*amoindrît*	V;SBJV;PST;3;SG
	enrober	*enrobé*	V.PTCP;PST
	approcher	*approcherait*	V;COND;3;SG
Georgian	ლენჩი	ლენჩებს	N;PL;DAT
	ჟანდარმერია	ჟანდარმერიათა	N;PL;LGSPEC2;ERG
	უსმენს	ვუსმენდით	V;1;PL;IND;IPFV
Greek	ξεχνώ	ξεχνάς	ADJV;2;SG;IPFV;PRS
	διαιρώ	να διαιρέσει	V;3;SG;PFV;SBJV
	χάνομαι	θα χαθείτε	V;2;PL;PFV;FUT
German	*einschließen*	*schlössest ein*	V;SBJV;PST;2;SG
	verbleiben	*verbliebt*	V;IND;PST;2;PL
	Adjektiv	*Adjektive*	N;GEN;PL
Russian	исход	исходами	N;INS;PL
	валлонский	валлонскому	ADJ;DAT;NEUT;SG
	усаживать	усаживаю	V;PRS;1;SG
Telugu	అమ్ము	అమ్ముతున్నది	V;3;FEM;SG;PRS;DUR
	అనుమానించు	అనుమానించారు	V;3;MASC;PL;PST
	అశోకుడు	అశోకులు	N;PL;NOM

Figure 3.11: An excerpt of data used in the experiment. There is no glossary given in the data.

and unproductive rules, the number of maximum rules grows rapidly from *low* to *medium* and then slows down to around twice as many from *medium* to *high*. For the detailed results of all languages please refer to Table D.10 in Appendix D.

3.6 Data augmentation

Preliminary results show that the neural approach suffers from data sparsity problems. To tackle this problem, we perform a simple data augmentation which artificially creates additional training data from pieces of evidence seen in the original training data. Additional training data is expected to bring improvement to the performance of our model, especially in *low* data conditions (Kann and Schütze, 2017; Bergmanis et al., 2017; Silfverberg et al., 2017; Zhou and Neubig, 2017; Nicolai et al., 2017). This idea was also proposed by (Irvine and Callison-Burch, 2014) to *hallucinate* additional entries in the phrase tables for statistical machine translation. Here we consider transducer-based rule extraction to hallucinate additional training data.

We search for the longest common substring between lemma and target form. The left part is assumed as a prefix candidate, while the right part is assumed a suffix candidate. Figure 3.13 shows several examples of rules extracted from the training data in three different languages.

3.6.1 Variants of the general affixing rules

To capture variants of an affixing rule where the next or previous character influences the changes, we added the first character from the longest common substring to the extracted prefix candidate and the last character for the suffix candidate. This, for example, happens for regular past form in English where you add only -*d* as a suffix for lemmata ending with *e*, instead of adding -*ed*. Another example is for the third singular present form in English where you add -*s* as a suffix for most lemmata; e.g. from *illustrate* to *illustrates*. However, for lemmata ending with -*ch*, -*s*, -*sh*, -*x* or -*z*, we use -*es* as their suffix; e.g. from *watch* to *watches*. Figure 3.12 shows an example of handling situational affixing for past tense inflection in English.

At a glance, it looks similar to the baseline system which is provided by SIGMORPHON (see Section 3.7.1). However, we only memorise the left (prefix candidate) and right part (suffix candidate), not all of the possible affix combinations with the stem as the baseline system does. It simplifies the rules extraction process, and thus, yields a smaller number of extracted rules in comparison to the baseline system.

	Prefix	Root	Suffix
Lemma		*wal**k***	
Target form		*wal**k***	***ed***

	Prefix	Root	Suffix
Lemma		*illustrat**e***	
Target form		*illustrat**e***	***d***

Figure 3.12: The first and last characters are remembered to handle situational affixing. Example given is for past tense inflection in English.

3.6.2 Creating additional training data

For each rule which appears less than 10 times in the training data, we artificially create 5 instances of additional training data. The additional training data is constructed by using a random string with a random length in the range of 1 to 4. Here, we do not employ any language model to assess the probability of the character sequence like the one described in (Silfverberg et al., 2017). For example, we create the following additional training instances for the examples given in Figure 3.13. Characters written in boldface come from prefixing and suffixing rules extracted from entries that exist in the data.

Irish: ***f**bsó**d*** $\Longrightarrow$ ***na f**bsó**dí***
French: ***a**if**rir*** $\Longrightarrow$ ***a**if**rît***
German: ***eins**raft**ließen*** $\Longrightarrow$ ***s**raft**lössest ein***

3.7 Experiments

We consider the use of three different approaches:

- morpheme-based (baseline),
- holistic (ours), and
- neural

For the neural approaches, we trained all of the systems on the *train* dataset and used the the *dev* dataset as the validation dataset for all the languages for all training dataset sizes. The performance of each system is then evaluated against the *test* dataset. For the morpheme-based and holistic approaches, we only exploit the *train* dataset. The *dev* dataset is not used. This can be seen as a disadvantage in comparison to the neural approaches. This issue is addressed in Section 3.9.

Insertion: Irish

		Entry	Prefix	Root	Suffix
Existing	**Lemma:**	*fótaidhé-óid*		*fótaidhé-óid*	
	Target MSD:	`N;NOM;PL;DEF`			
	Target form:	*na fótaidhé-óidí*	*na*	*fótaidhé-óid*	*í*
Extracted				***f … d***	
			na	***f … d***	***í***
Generated	**Lemma:**	***fbsód***		***fbsó****d*	
	Target MSD:	`N;NOM;PL;DEF`			
	Target form:	***na fbsódí***	*na*	***fbsó****d*	*í*

Substitution: French

		Entry	Prefix	Root	Suffix
Existing	**Lemma:**	*amoindrir*		*amoindr*	*ir*
	Target MSD:	`V;SBJV;PST;3;SG`			
	Target form:	*amoindrît*		*amoindr*	*ît*
Extracted				***a … r***	***ir***
				a … r	***ît***
Generated	**Lemma:**	***aifrir***		***aif****r*	*ir*
	Target MSD:	`V;SBJV;PST;3;SG`			
	Target form:	***aifrît***		***aif****r*	*ît*

Deletion and substitution: German

		Entry	Prefix	Root	Suffix
Existing	**Lemma:**	*einschließen*	*ein*	*schl*	*ießen*
	Target MSD:	`V;SBJV;PST;2;SG`			
	Target form:	*schlössest ein*		*schl*	*össest ein*
Extracted			***ein***	***s … l***	***ießen***
				s … l	***össest ein***
Generated	**Lemma:**	***eins****raft****ließen***	*ein*	***sraft****l*	*ießen*
	Target MSD:	`V;SBJV;PST;2;SG`			
	Target form:	*sraft****lössest ein***		***sraft****l*	*össest ein*

Figure 3.13: Illustrations of rules extracted for data augmentation: simple insertion (Irish); substitution (French); deletion and substitution at the same time (German).

substring	replacement	# of occurrences
'-ε'	*'-ing'*	1,121
'-e'	*'-ing'*	832
'-ize'	*'-izing'*	162
⋮	⋮	⋮
'show'	*'showing'*	1
⋮	⋮	⋮

Figure 3.14: Illustration of affixes remembered by the baseline system from the training data. It memorises all the differences between the word form and the lemma in various character length and their number of occurrences in the training data.

3.7.1 Baseline: morpheme-based

The SIGMORPHON 2018 Shared Task provided a baseline system (Cotterell et al., 2018) for morphological generation task which adopts a morpheme-based approach. The system initially learns all of the affixes from the training data and subsequently leverages the rules to generate the predicted target form. Furthermore, the system assesses whether a given language is biased towards either prefixing or suffixing. If the language favours prefixing, the string is reversed to adhere to this preference.

3.7.1.1 Learning

In the morpheme-based approach, each instance in the training data is analysed using the Levenshtein distance to align the lemma and the word form. It is then used to break down the word into three parts: prefix, stem, and suffix. These affixing rules are grouped based on the given target MSD. They are stored as knowledge in a list of triplets: the substring to be replaced, the substring with which it is to be replaced, and the number of occurrences of this rule in the training dataset. Figure 3.14 illustrates suffixing rules stored in the system for English. For example, the system observed 832 occurrences of the '-*e*' substring being replaced with '-*ing*' in the given dataset for the present participle.

3.7.1.2 Generation

In the generation step, it filters the candidate rules by the given target MSD. First, the longest common suffixing rule with the highest number of occurrences is applied. Then the most frequent prefixing rule is applied in succes-

Training data

Lemma	**Target form**	**Target MSD**
age	*ages*	`V;3;SG;PRS`
age	*aged*	`V;PST`
watch	*watches*	`V;3;SG;PRS`
watch	*watched*	`V;PST`
revise	*revises*	`V;3;SG;PRS`
⋮	⋮	⋮

Question

Lemma:	*illustrate*
Target MSD:	`V;3;SG;PRS`

Figure 3.15: An example of given training data and question in English

sion to generate the predicted target form. If the given target MSD is not found in the training data, the system will return the lemma as the answer.

3.7.2 Holistic: generating target form using analogical grids

In contrast to the baseline system which uses a morpheme-based approach, Instead of breaking words into pieces, which is used in the morpheme-based approach, we take a holistic approach (Singh, 2000; Singh and Ford, 2000; Neuvel and Singh, 2001). We generate the target form by solving analogical equations based on the evidence observed in the given training data.

Let us say that we are given a set of training data (*left*) and a question (*right*) as shown in Figure 3.15. First, we extract all of the analogical grids from the given training data. The characters and MSD are used as dimensions for the word vector representation to take into account both the level of form and morphology of the word. Then, the relevant analogical grid is selected according to the given target MSD. If several candidates of analogical equations exist, we use some heuristic features to select the analogical equation. These heuristics are edit distance, the longest common subsequence, the longest common suffix, and the longest common prefix, between the given lemma and lemmata existing in the training dataset. If there are still several candidates after using heuristic features, we solve all of the possible analogical equations to generate all the possible predicted target forms. The most frequent answer is chosen as the predicted target form.

Figure 3.16 illustrates how to generate the target form for the example given in Figure 3.1. Let us say that we are able to get two analogical grids according to the given MSD. We construct the analogical equation as follows:

LEMMA	:	V;3;SG;PRS	:	V;PST
age	:	*ages*	:	*aged*
revise	:	*revises*	:	*revised*
compare	:	*compares*	:	*compared*
bake	:	*bakes*	:	

$$\textit{age} : \textit{ages} :: \textit{illustrate} : x \\ \Rightarrow \\ x = \boldsymbol{illustrates}$$

LEMMA	:	V;3;SG;PRS	:	V;PST
watch	:	*watches*	:	*watched*
miss	:		:	*missed*
publish	:	*publishes*	:	*published*
fetch	:		:	*fetched*

$$\textit{watch} : \textit{watches} :: \textit{illustrate} : x \\ \Rightarrow \\ x = \cancel{\textit{illustratees}}$$

Figure 3.16: How to generate target form (3rd person singular present) of the given lemma *illustrate* by solving analogical equation. Different analogical grids may generate different target forms. The analogical grid on the top produces the form *illustrates*, while the analogical grid on the bottom produces the form *illustratees*.

$$\text{lemma}_t : \text{form}_t :: \textit{illustrate} : \text{form}_q$$

taken from the first and second columns of the analogical grids according to the given MSD. Based on the longest common suffix, we choose to use the analogical grid on the top which produces the word form *illustrates* instead of the bottom one which produces *illustratees*.

$$\textit{age} : \textit{ages} :: \textit{illustrate} : x \quad \Rightarrow x = \boldsymbol{illustrates}$$
$$\textit{watch} : \textit{watches} :: \textit{illustrate} : x \quad \Rightarrow x = \cancel{\textit{illustratees}}$$

There is an issue that is similar to the baseline system. If the given target MSD is never seen in the training data, the system will output the lemma as the default output. One may try to loosen the constraint on the target MSD. The idea is to find the most similar target MSD(s) that have been seen in the training data if we are given an unseen target MSD. Instead of just returning the lemma, we find the most similar target MSD and use it to generate the target form. This is based on the assumption that similar target MSDs probably share the same affixing phenomena. Similar target MSD can be selected using the longest common subsequence or highest recall score. We can also introduce weighting on the MSFs to differentiate which MSFs are more *decisive* on the affixing rule in comparison to the other MSFs inside a target MSD. (Kuroda, 2016) shows how to use Formal Concept Analysis to explain how morphological features (MSDs in our case) influence the construction of the inflectional paradigm in the mind of Czech speakers for declensions in Czech.

3.7.3 Neural approach

Following the recent success of the neural approach in previous evaluation campaigns, we implement a common architecture of the sequence-to-sequence (seq2seq) model. We treat the inflection task as the problem of translating the given target MSD and lemma into target form.

$$\mathrm{MSF}_1 \quad \mathrm{MSF}_2 \quad \ldots \quad \mathrm{MSF}_\mathrm{m} \quad \mathrm{c}_1 \quad \mathrm{c}_2 \quad \ldots \quad \mathrm{c}_\mathrm{n}$$

We feed the sequence of MSFs followed by the characters of the given lemma into the system. Thus, the input string for the example given in Figure 3.1 is as follows.

$$V \quad 3 \quad SG \quad PRS \quad i \quad l \quad l \quad u \quad s \quad t \quad r \quad a \quad t \quad e$$

3.7.3.1 Sequence-to-sequence model

Our model is a standard sequence-to-sequence (seq2seq) model with an attention mechanism inspired by the one which is used for machine translation (Luong et al., 2015). The difference is that we consider a character or MSD as one token, instead of a word. Each token (character) is represented by a continuous vector representation learned in the embedding layer. Figure 3.17 shows the architecture of the neural network used in this work.

We use a bi-directional Gated Recurrent Unit (GRU) cell (Cho et al., 2014) which is a variation of the Long Short-Term Memory (LSTM) cell (Hochreiter and Schmidhuber, 1997) that tries to solve the vanishing gradient problem. Our decoder has two layers of uni-directional GRU cells with an attention mechanism. There are various implementations of attention mechanism like (Bahdanau et al., 2015; Luong et al., 2015). In this work, we use the one that has weight normalization (Salimans and Kingma, 2016) to help the model converge faster.

To handle unseen tokens, we remember them in a First-In-First-Out (FIFO) list and replace them with a special token *<UNK>* before feeding them into our model. These special tokens are reverted to the character contained in the list after the decoding phase.

3.7.3.2 Hyperparameters

We fixed our hyperparameters for all languages and amounts of resources after doing some preliminary experiments. The number of hidden units is

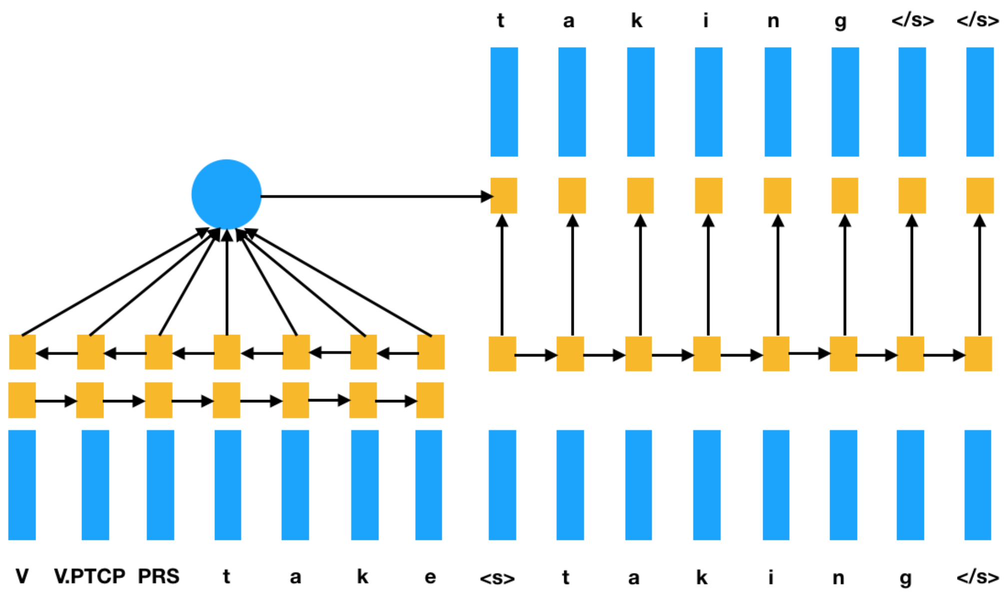


Figure 3.17: Seq2seq model using bi-directional LSTM encoder and uni-directional LSTM decoder with attention mechanism

fixed to 100 for each layer in the encoder and decoder. The size of the embedding is 300. We optimise the model using ADAM (Kingma and Ba, 2015) with a learning rate of 5×10^{-4} during training. To make the training process faster, we use a mini-batch size of 20.

We trained the model using an early-stop mechanism of 30 epochs without improvement on validation data which is the *dev* dataset.

3.7.3.3 Analogical grid as a feature

We can enforce the model to remember that the current training instance is related to the other by introducing analogical grid as a feature. For every training instance, we added the identifiers (ID) of the analogical grid containing the lemma or the target MSD of the current training instance.

$$\text{ID} \quad \text{MSF}_1 \quad \text{MSF}_2 \quad \ldots \quad \text{MSF}_m \quad \text{c}_1 \quad \text{c}_2 \quad \ldots \quad \text{c}_n$$

For example, the input of the system will be like this:

GRID1 V 3 SG PRS i l l u s t r a t e

3.7.3.4 Few-shot model

The multi-source model is a way to train a single model for several languages at once. (Zoph et al., 2016; Johnson et al., 2017; Neubig and Hu, 2018; Aharoni et al., 2019) showed that training several languages at the same time allows for a single model to perform the task in multiple languages. This approach is useful when there is scarcity in the data, where some unseen classes exist.

$$\mathrm{LANGCODE}\quad \mathrm{MSF}_1\quad \mathrm{MSF}_2\quad \ldots\quad \mathrm{MSF}_\mathrm{m}\quad \mathrm{c}_1\quad \mathrm{c}_2\quad \ldots\quad \mathrm{c}_\mathrm{n}$$

In addition to the seq2seq model, we also perform experiments with a model trained on all of the languages at once. To perform these experiments, we use codes to represent the current language, for example: *¡en¿* means that the current entry is English. The language code of an input string is added to the beginning of the input sequence.

<en> V 3 SG PRS i l l u s t r a t e

We perform two sets of experiments: The first one is to train using all of the data at once, and the second one is to divide the training data into groups of language that belong to the same language family.

3.7.4 Hybrid approach

Several systems are developed for the task. The first one is based on a holistic approach. We generate the target forms by solving analogical equations on words. The second one is a seq2seq neural network model. Simple data augmentation is also implemented to help in low-resource conditions. We evaluated their performance on the development dataset and chose the best system on each language and dataset size as our representative system for the hybrid system. This system is then tested against the test dataset.

3.7.5 Evaluation metrics

We evaluate the performance of the systems on each language against 2 measures: accuracy and average Levenshtein distance. While accuracy demands a strict evaluation (all or nothing), the average Levenshtein distance offers a more relaxed evaluation which helps us understand how much the prediction differs from the answer (in characters).

3.7.5.1 Accuracy

Accuracy is the ratio of correctly predicted target forms by the total number of questions. In this metric, the higher the score, the better. Formula (3.6) gives the exact definition[1].

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} \delta(\text{predicted}_i = \text{correct}_i)}{N} \times 100 \tag{3.6}$$

3.7.5.2 Average Levenshtein distance

Average Levenshtein distance is the average of all Levenshtein distance values over the questions[2]. It is used to measure how close the prediction is to the true answer. Here, the lower the score, the better.

$$\text{Average Levenshtein distance} = \frac{\sum_{i=1}^{N} lev(\text{predicted}_i, \text{correct}_i)}{N} \tag{3.7}$$

3.8 Results and analysis

Table 3.4 shows the average results across all of the 103 languages. Please refer to Table D.1 in Appendix D for more detailed results in each language.

The holistic approach outperforms the baseline system under all conditions (*low*, *medium* and *high*). Furthermore, it achieves the best accuracy in comparison to the morpheme-based and neural approaches under *low* data conditions. On top of that, our holistic approach even achieved the smallest average Levenshtein distance under *high* data conditions. This means that, in comparison to other approaches, our holistic approach outputs the closest prediction to the answer even when it gives the wrong answer.

The results show that the neural approach using the seq2seq model left behind both the baseline system and the holistic approach in *medium* and *high* data conditions. The gap is around 15% of accuracy. However, the lack of training data exhibits the drawback of the neural approach as it performs poorly in *low* data conditions. Furthermore, the use of data augmentation improves performance in most cases. We can see an improvement of around 3 times better accuracy on the *low* dataset although it still cannot overcome

[1] N is the total number of questions. $\delta(A = B)$ equals to 1 if the two strings A and B are same, else it is 0.

[2] $lev(A, B)$ is the Levenshtein distance between strings A and B, described in (Levenshtein, 1966; Wagner and Fischer, 1974). It equals to 0 if the two strings A and B are same, or else it is the edit distance between them.

Table 3.4: Average accuracy scores on *test* dataset.

Method	**Accuracy**			**Levenshtein distance**		
	low	**medium**	**high**	**low**	**medium**	**high**
Morpheme	38.3	61.8	74.7	1.9	1.0	0.6
Holistic (ours)	39.6	64.1	77.2	2.0	1.0	**0.1**
Seq2seq	13.1	71.3	90.9	2.3	0.9	0.2
Seq2seq+Aug	36.9	**78.5**	89.1	2.1	**0.7**	0.2
Few-shot	25.5	67.6	71.3	2.3	0.8	1.2
Hybrid	**44.1**	77.4	**91.1**	**1.7**	**0.7**	0.2

the performance of both the baseline and the holistic approach. However, we found that the multi-source model seems to struggle on the *high* dataset. Our intuition is that the model encounters difficulties in converging because of the large amount of data and the size of the model. Using a bigger model probably helps to improve the performance.

The baseline system and the holistic approach shine over the neural approach, particularly for languages like Albanian, Czech, English, French, Haida, Neapolitan, Bokmaal (Norwegian), Quechua, and Uzbek (see Table 3.5). Our seq2seq model seems to struggle even on the *high* dataset for some of these languages. On the other hand, our seq2seq model gets better accuracy than the baseline system or holistic approach even on the *low* dataset in some languages like Azeri, Basque, Breton, Cornish, Greenlandic, Hindi, Karelian, Khaling, Maltese, Middle-Low-German, Middle-High-German, Murrinhpatha, Norman, North-Frisian, Persian, Swahili, Turkish, Turkmen, Welsh, Zulu.

The same trend can be seen in the results for similar languages, like Romance (Catalan, Galician, Portuguese, and Spanish), Semitic (Arabic and Hebrew), and Baltic (Latvian and Lithuanian) languages. The baseline system leads the score on *low* dataset size before starting to be outperformed by our seq2seq model on the dataset with bigger sizes (see Table 3.6). For other language families like Indo-Aryan (Hindi, Urdu), Finnic (Estonian and Finnish), and Turkic (Turkish) languages, our seq2seq model steadily leads the score for all dataset sizes (see Table 3.7).

Table 3.5: Accuracy scores on languages where morpheme-based system (**M**) and holistic approach (**H**) perform better than neural approach: seq2seq model without data augmentation (**S**) and with data augmentation (**S-Aug**).

Language	**Accuracy**											
	low				**medium**				**high**			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
albanian	5.8	23.8	0.6	11.6	13.2	72.1	44.8	65.2	12.5	88.3	81.1	80.5
czech	38.5	38.5	1.6	26.1	79.9	81.7	51.1	76.6	90.6	90.8	85.5	86.3
haida	29.0	14.0	5.0	23.0	61.0	62.0	50.0	52.0	66.0	59.0	53.0	52.0
neapolitan	79.0	74.0	25.0	65.0	94.0	93.0	91.0	95.0	95.0	95.0	95.0	95.0
norwegian-bokmaal	67.8	72.2	13.8	54.8	80.7	82.4	78.0	76.5	91.0	90.0	88.9	77.0
quechua	15.9	10.3	3.2	31.2	70.9	50.4	52.0	55.9	95.1	89.1	56.3	56.0
uzbek	52.0	30.0	47.0	74.0	96.0	93.0	78.0	78.0	96.0	93.0	78.0	78.0
english	77.6	83.2	28.5	56.4	90.5	91.4	85.7	88.0	95.9	95.4	95.6	93.6
french	59.0	56.6	3.9	37.7	73.2	72.1	71.9	71.6	83.0	83.0	83.7	73.5

Table 3.6: Accuracy scores on languages where morpheme-based system (**M**) and holistic approach (**H**) perform better on *low* dataset before outperformed on bigger dataset by the neural approach: seq2seq model without data augmentation (**S**) and with data augmentation (**S-Aug**).

Language	**Accuracy**											
	low				**medium**				**high**			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
arabic	26.8	27.9	0.1	21.0	39.5	48.6	61.1	67.9	47.0	62.5	93.0	91.7
armenian	37.0	33.2	1.2	34.2	70.4	77.8	76.5	83.7	86.6	88.0	94.1	90.9
bengali	50.0	49.0	14.0	49.0	76.0	74.0	94.0	96.0	81.0	83.0	98.0	99.0
catalan	60.8	57.1	4.6	32.6	85.6	83.9	85.0	92.3	95.7	94.6	98.1	95.9
classical-syriac	94.0	92.0	41.0	72.0	99.0	99.0	94.0	98.0	97.0	96.0	98.0	100.0
crimean-tatar	56.0	67.0	16.0	63.0	78.0	80.0	95.0	89.0	95.0	93.0	99.0	98.0
danish	58.3	64.9	30.2	53.0	77.8	79.1	74.3	69.8	87.0	86.5	91.3	85.8
faroese	34.4	39.2	3.3	16.6	65.2	68.1	51.0	60.6	76.1	76.6	79.8	74.5
friulian	70.0	71.0	25.0	49.0	92.0	92.0	89.0	94.0	96.0	97.0	98.0	99.0
galician	53.0	51.1	9.1	30.7	82.8	81.5	77.9	88.9	95.1	94.6	98.4	97.4
georgian	70.6	68.8	17.2	58.9	92.1	91.6	82.9	92.5	93.9	93.7	98.5	98.4
greek	13.6	24.4	2.0	12.0	15.2	59.9	44.3	56.6	16.5	77.6	81.7	83.3
hebrew	24.4	25.5	4.1	13.8	38.1	50.1	76.3	76.3	53.7	61.7	98.1	97.2
hungarian	17.4	27.5	0.9	12.1	44.4	51.1	47.3	53.1	68.8	69.5	77.5	63.5
ingrian	20.0	26.0	27.5	20.0	46.0	50.0	80.0	75.0				
irish	31.6	34.2	3.7	20.9	37.0	48.9	42.6	57.7	39.4	60.1	83.0	77.2

Continued on next page

Table 3.6 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
italian	40.5	42.2	3.3	41.3	72.5	84.8	81.3	91.1	77.5	94.4	97.9	95.4
kashubian	60.0	50.0	12.5	57.5	68.0	56.0	85.0	92.5				
kurmanji	82.7	86.5	0.0	58.4	85.2	88.4	83.7	88.2	92.9	91.9	92.8	91.4
ladin	58.0	52.0	30.0	52.0	86.0	83.0	88.0	95.0	92.0	92.0	98.0	98.0
latin	16.0	14.5	0.8	5.4	37.6	29.7	25.2	36.2	47.6	39.7	70.1	55.5
latvian	52.2	48.0	4.1	18.3	85.5	88.2	60.5	82.4	92.8	93.1	94.8	94.8
lithuanian	23.3	18.3	0.8	5.6	52.2	49.4	33.7	51.6	64.2	64.0	86.2	84.1
livonian	28.0	29.0	1.0	27.0	51.0	53.0	69.0	77.0	67.0	66.0	92.0	92.0
lower-sorbian	32.1	36.9	2.9	19.3	68.9	79.1	64.1	81.4	88.1	87.5	95.2	94.8
macedonian	49.8	45.2	5.1	37.7	82.6	85.3	75.7	89.8	91.2	92.1	96.4	95.3
middle-french	76.9	75.7	10.1	67.2	90.3	90.9	89.2	93.0	95.1	93.6	98.8	96.3
navajo	16.6	16.8	2.0	13.8	30.4	29.1	35.8	41.5	39.0	37.7	82.5	76.0
northern-sami	16.4	11.4	2.1	11.6	34.8	32.8	43.2	60.7	62.3	61.5	93.4	88.0
norwegian-nynorsk	48.9	56.0	11.9	37.6	61.1	62.7	52.5	57.0	74.8	73.7	84.0	75.8
occitan	72.0	69.0	15.0	55.0	92.0	87.0	94.0	98.0	96.0	93.0	100.0	100.0
old-armenian	31.0	30.4	1.5	14.8	67.3	70.8	48.9	69.3	79.2	81.1	86.0	85.1
old-church-slavonic	39.0	39.0	11.0	29.0	76.0	71.0	74.0	78.0	80.0	70.0	92.0	96.0
old-saxon	22.8	16.0	2.7	5.2	39.0	34.5	63.0	68.0	60.1	52.9	95.3	94.6
pashto	35.0	33.0	8.0	21.0	69.0	65.0	69.0	75.0	72.0	70.0	100.0	98.0

Continued on next page

Table 3.6 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
portuguese	62.6	61.7	6.9	31.0	92.4	91.2	78.2	92.5	96.7	96.3	97.6	97.5
romanian	44.8	42.5	3.2	30.3	69.4	71.1	59.7	72.3	79.8	77.4	84.6	83.1
slovak	37.7	48.1	3.3	23.8	71.1	73.5	61.3	70.6	83.1	81.8	90.0	89.9
slovene	32.3	34.1	13.7	25.9	72.3	73.5	63.4	86.0	85.1	83.5	95.2	93.8
sorani	19.3	17.9	1.2	15.6	51.7	49.1	60.3	71.4	63.6	62.8	88.0	87.7
spanish	61.8	57.4	4.9	46.7	86.3	85.7	84.3	90.3	92.4	94.8	97.1	95.8
swedish	51.1	60.8	7.8	39.9	76.5	77.3	62.2	68.0	84.7	84.4	86.1	76.2
tatar	52.0	72.0	17.0	53.0	89.0	89.0	94.0	87.0	95.0	95.0	100.0	99.0
turkmen	34.0	68.0	37.5	60.0	68.0	74.0	87.5	92.5				
ukrainian	38.7	46.9	6.7	23.3	74.1	74.7	55.3	71.3	86.3	84.8	89.9	87.1
venetian	71.8	71.3	16.6	42.3	89.1	87.3	91.6	93.1	93.0	91.6	99.6	99.0
west-frisian	50.0	46.0	8.0	40.0	65.0	61.0	86.0	93.0	67.0	63.0	91.0	95.0
yiddish	78.0	79.0	6.0	60.0	87.0	88.0	83.0	92.0	94.0	86.0	98.0	99.0
dutch	50.8	53.5	7.8	24.1	72.4	74.0	73.5	79.4	87.7	86.8	96.2	95.1
german	49.2	50.9	10.7	11.5	71.7	74.2	66.0	71.1	81.1	81.6	88.4	82.0
kannada	33.0	36.0	9.0	27.0	55.0	63.0	83.0	90.0	66.0	66.0	95.0	95.0
polish	40.4	41.3	1.8	13.9	73.5	76.1	60.0	76.1	87.1	87.1	88.1	89.5
russian	43.4	46.1	1.8	11.5	76.4	79.7	54.4	76.5	86.5	87.6	89.2	87.7

Table 3.7: Accuracy scores on languages where the neural approach: seq2seq model without data augmentation (**S**) and with data augmentation (**S-Aug**) steadily outperforms morpheme-based system (**M**) and holistic approach (**H**) on all dataset sizes

Language	**Accuracy**											
	low				**medium**				**high**			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
estonian	21.5	19.2	0.7	28.4	62.9	61.1	60.0	70.3	78.0	78.0	90.6	88.0
finnish	10.4	16.1	0.7	18.7	44.1	41.5	42.6	69.9	78.0	76.6	84.1	82.0
hindi	31.8	28.6	23.9	65.6	86.5	85.2	94.3	95.1	93.0	92.0	98.6	97.5
turkish	13.2	11.3	1.1	28.5	32.8	41.8	71.4	68.3	73.2	75.4	91.8	87.0
urdu	32.7	29.6	24.9	57.8	87.6	85.6	91.5	95.0	95.9	94.7	97.4	97.6

3.9 Discussion

The results for the baseline system and our holistic approach show the absence of the necessity to break down words into morphemes. The derivation between lemma and target form can also be acquired through analogy. However, selecting the candidates for constructing the analogical equation is crucial thing. Thus, we need to improve our selection method or use better heuristic features. To handle the problem of unseen MSD, the use of formal concept analysis (Ganter and Wille, 1999) is worth considering.

3.9.1 How much data is enough?

The improvement shown by using data augmentation seems promising. One may think of increasing the amount of artificially created additional training data. However, there is a trade-off between performance and training time. More training data means more space to search for the baseline and the holistic approach. For the neural approach, more training data requires more time to train.

Another thing to consider is how much more additional training data should be created. We can see that the data augmentation seems not to improve the performance in *high* data conditions anymore. Figure 3.18 shows the performance of the systems against the size of training data. We can see that the neural approach starts with poor performance on training data with small sizes. The neural approach starts to outperform the baseline and holistic approach when the training data contains around 700 samples.

This thing is also shown by the performance of the holistic approach. We performed additional experiments to see whether having more data will improve the performance of the holistic approach. We give the *dev* dataset as supplementary data along with the *train* dataset, noted as Holistic+`dev`. Table 3.8 shows the comparison between Holistic and Holistic+`dev`. We can observe that it significantly improves the performance of the system in low-resource conditions. However, after some point, the *dev* dataset gives no more improvement. It starts when the size of the *train* dataset is 5,000 samples (see Figure 3.18). If we focus on the *low* data conditions, our Holistic+`dev` outperforms all the other approaches. It even beats the SIGMORPHON 2018 Shared Task winner, UZH-02. For the full comparison with other systems submitted to the SIGMORPHON 2018 Shared Task, please refer to Table D.2 in Appendix D.

Another concern when creating more data is that it may change the ratio of frequency between regular and irregular forms that exist in the data. If our goal is to have a model that is capable of generalising over the regular forms

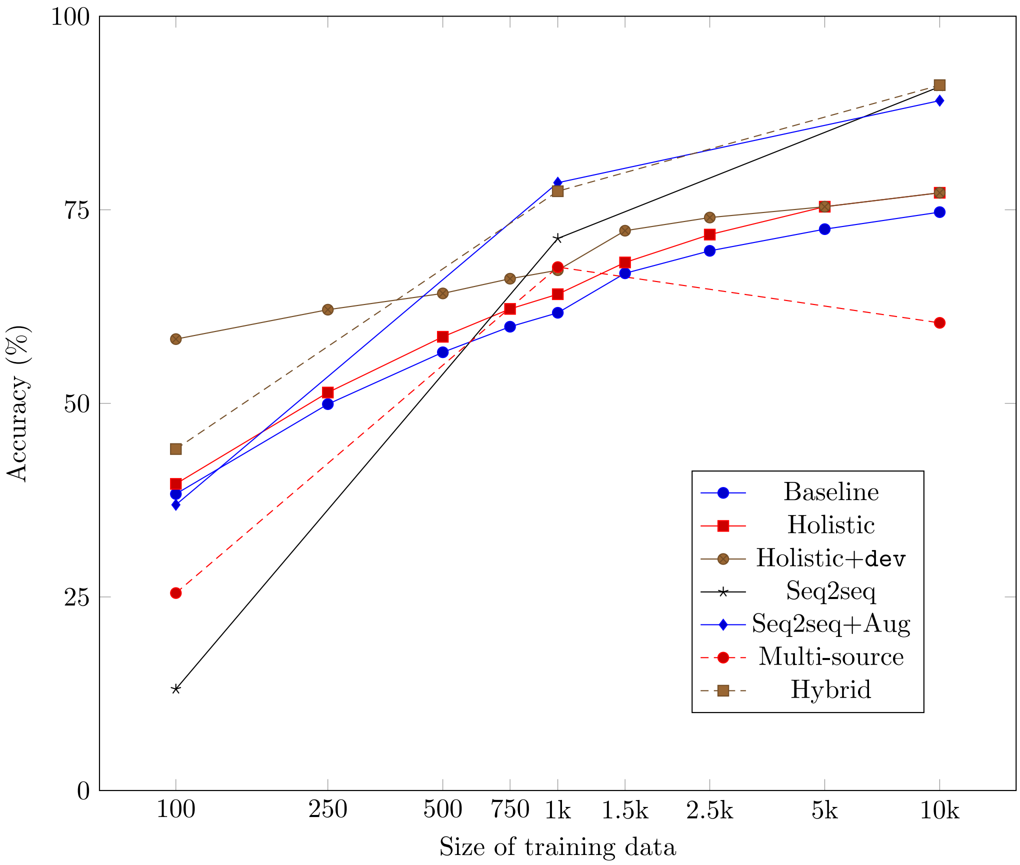


Figure 3.18: Performance of systems trained on different sizes of dataset. Caution: log scale on the x-axis.

Table 3.8: Comparison of average accuracy and Levenshtein distance scores on *test* dataset between holistic approach with and without the help of *dev* dataset.

Method	**Accuracy**			**Levenshtein distance**		
	low	**medium**	**high**	**low**	**medium**	**high**
Morpheme	38.3	61.7	74.7	1.9	1.0	0.6
Seq2seq	13.1	71.3	90.9	2.3	0.9	0.2
Few-shot	25.5	67.6	60.4	2.3	0.8	1.2
Seq2seq+Aug	36.9	78.5	89.1	2.1	0.7	0.2
Holistic (ours)	39.6	64.1	77.2	2.0	1.0	0.1
Hybrid	44.1	77.4	91.1	1.7	0.7	0.2
UZH-02	57.2	86.4	96.0	1.0	0.3	0.1
Holistic+`dev` (ours)	58.3	67.2	77.2	1.3	0.9	0.1

but also aware of the irregular ones, one should be more cautious to *preserve* the ratio that emerges from the original data. In this case, the statistics given by the Tolerance Principle (remember Section 3.5.3) may help to keep the proportion between regulars and irregulars form.

3.9.2 Analogy for data augmentation

Section 3.6 showed how the transducer-based rule extraction is used to create additional training data. This method to extract the affix rules is very simple. Although it may capture circumfixes, it is still strongly biased to prefixing and suffixing only. A better method is expected to also capture other phenomena, such as parallel infixing (Arabic), reduplication (Greek), and repetition (Malay and Indonesian). We consider analogy as another possible way to create more training data. By treating the word as a whole, we expect to capture more morphological phenomena.

3.9.3 Morphological complexity of the language

Languages belonging to the same family are expected to exhibit similar morphological phenomena. (Bentz et al., 2016) provides a study of morphological complexity in more than 500 languages of 101 language families. One of the measures presented is C_{WALS}. It simply computes the average of feature value f over the number of features n of a given language.

Table 3.9: Overview of morphological complexity for all data sizes computed using Formula (3.8).

Data	$\sum_{i=1}^{n} f_i$			n			C_{WALS}		
	Min	**Avg**	**Max**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**
low	1	19	36	1	8	14	0.052	0.280	0.451
medium	1	21	38	1	8	15	0.053	0.300	0.466
high	1	22	38	1	9	15	0.053	0.310	0.466

Table 3.10: Overview of estimated morphological complexity for all data sizes computed using Formula (3.8).

Data	$\sum_{i=1}^{n} f_i$			n			estimated C_{WALS}		
	Min	**Avg**	**Max**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**
low	5	22	43	2	6	11	1.75	3.97	8.40
medium	5	23	48	2	6	14	1.75	4.16	9.20
high	7	25	48	3	6	14	1.75	4.29	9.20

$$C_{\text{WALS}} = \frac{\sum_{i=1}^{n} f_i}{n} \tag{3.8}$$

To have comparable numbers across languages, the feature value is normalised against the maximum number of feature values that can be used for each language. Table 3.9 shows the overview of the morphological complexity of each dataset size. For the detailed results of all languages please refer to Table D.8 in Appendix D.

Unfortunately, the complete schema of the dataset has not yet been published on the Unimorph Project page. There are some MSFs that are not listed in the schema (e.g. `NDEF`, `LOC`, etc.). Due to this reason, we modify Formula (3.8) and estimate the number of features by the longest MSD found in the dataset for each language. In this setting, we define our $\sum_{i=1}^{n} f_i$ as the total number of unique MSFs, and estimate n with the longest MSD (count by MSF) found in the dataset. Table 3.10 shows the overview of the estimated morphological complexity for each dataset size. English has the lowest C_{WALS} score. Quechua, on the other hand, has the highest C_{WALS} score on all *low*, *medium* and *high* datasets. For the detailed results of all languages please refer to Table D.9 in Appendix D.

Pearson and Spearman correlation coefficient is calculated to observe whether there is a correlation between the morphological complexity of a

Table 3.11: Pearson and Spearman correlation coefficient between C_{WALS} and system's accuracy. ‡ stands for $p < 0.05$, while the other p-value range from 0.1 to 0.9.

Method	**Pearson**			**Spearman**		
	low	**medium**	**high**	**low**	**medium**	**high**
Baseline	-0.01	0.04	0.14	-0.07	0.04	0.14
Holistic	-0.04	0.04	0.14	-0.09	0.06	0.13
Seq2seq	-0.31‡	-0.09	0.30‡	-0.29‡	-0.09	0.22‡
Seq2seq+Aug	-0.15	0.02	0.27‡	-0.14	0.03	0.25‡

language and the performance achieved by our system. Table 3.11 shows the correlation between the C_{WALS} score and system performance. From the correlation coefficient, we found that most of them show that there is a very low correlation between the baseline and holistic systems. We can observe a higher correlation for neural approaches but the correlation coefficients still vary too much along the dataset size, especially on the *medium* dataset. In this case, also considering the p-value for being too high, one may assume that there may be no correlation at all.

Let us now turn to Table 3.12 where the correlation is computed on the estimated C_{WALS} instead of using the real formula. Again, we found a very low correlation between the baseline and holistic systems. However, a higher correlation coefficient (with a very low p-value) can be observed for the neural approach. We think that it is probably influenced by how the neural approach works. The target MSD (sequence of MSF) and lemma are given as a sequence of input to the neural approaches, without any knowledge about how MSFs may belong to a specific feature group. In summary, the system treats the MSD as a long sequence of MSF and reads them one by one. This relates to how we compute the estimation of C_{WALS}, where we use the longest MSD found in the dataset to approximate the number of feature groups. The neural approaches do not differentiate the MSFs into groups as inputs so the length of the input, in this case, the length of the MSD, may influence the performance of the neural approach systems. This may explain why the estimated C_{WALS} using the longest MSD gives a higher correlation instead of the true C_{WALS}.

Table 3.12: Pearson and Spearman correlation coefficient between estimated C_{WALS} and system's accuracy. ‡ stands for $p < 0.05$, while the other p-value range from 0.1 0.6.

Method	**Pearson**			**Spearman**		
	low	**medium**	**high**	**low**	**medium**	**high**
Baseline	-0.23‡	-0.04	0.06	-0.22‡	-0.14	-0.09
Holistic	-0.25‡	-0.06	0.05	-0.23‡	-0.15	-0.11
Seq2seq	-0.31‡	-0.38‡	-0.42‡	-0.29‡	-0.43‡	-0.42‡
Seq2seq+Aug	-0.35‡	-0.29‡	-0.38‡	-0.38‡	-0.34‡	-0.37‡

3.10 Summary of the chapter

We studied the advantage of organising lexica for morphological inflection task. Word forms are organised as analogical grids to generate the target form by exploiting the neighbouring word forms found in grids. Experimental results show that we outperform the baseline. Our holistic approach always performs better than the baseline system on all sizes of the training dataset, from *low* to *high*. From the point of view of affixes, we observed that by treating the words as a whole, we are able to capture more affixing phenomena. We are also able to gain an improvement of around 6% in accuracy (up to 8%) in low-resource conditions by using a hybrid approach.

Focusing on the *low* resource conditions, our holistic approach improves the accuracy by 1.3% without the *dev* dataset and by 20% with the *dev* dataset in comparison to the morpheme-based approach. In addition, our holistic approach outperforms the winner system of the 2018 Shared task by 1.1%. It outperforms the winner system in 60 out of the total of 103 languages. We also found that under high-resource conditions, our holistic approach outperforms other methods with 0.1 in edit distance and outputs word forms that are closer to the answers. On average, our proposed method, Holistic+`dev`, achieves the best performance in morphologically rich languages under *low* resource conditions.

We further analysed the advantage of data augmentation which improves the performance of the neural approach. However, we found that, after some point, data augmentation does not improve the performance anymore and might lead to lower performance instead. We investigated the correlation between systems' performance and the morphological complexity of languages. We estimated the complexity of the language by using the C_{WALS} score and computing the Pearson and Spearman correlation coefficient. The numbers showed that there might be some correlation (with a very low p-value) specifically for the performance of the neural approach.

From our experimental results, we state the following main findings.

- The holistic approach has similar, or even slightly better, performance in comparison to the morpheme-based approach which proves our hypothesis on the absence of necessity to break down words into morphemes.
- The holistic approach outperforms the morpheme-based and neural approaches under low-resource conditions.
- Data augmentation helps the neural approach to gain improvement (around 3 times better accuracy) under low-resource conditions.
- The hybrid approach achieves the best performance under almost all conditions.
- There is a correlation between the complexity of the language with the performance of neural approaches which is probably caused by how the dataset was set up, i.e., very much biased towards machine learning approaches with training, development, and test sets.

Finally, the approaches are exploited for the opposite direction of the task, which is morphological analysis. This is addressed in Chapter 4.

Chapter 4

Morphological analysis using analogical grids

In this chapter, we apply the concept of analogical grids to the task of morphological analysis. This task is the reciprocal task of morphological generation (See Chapter 3). It consists of two main subtasks, lemmatisation and MSD analysis.

4.1 Organisation of the chapter

This chapter is organized as follows: Section 4.3 describes the morphological analysis task and summarises the contributions of the chapter. Section 4.4 presents the data used for the experiments. Section 4.5 introduces our experimental protocols and evaluation metrics. Section 4.6 shows the experimental results. It also analyses the results for both lemmatisation and MSD analysis. Section 4.7 presents a discussion and further experiments regarding the obtained experimental results. Section 4.8 gives a summary.

4.2 List of publications

The research described in this chapter has been published in the following publications[1].

Conference paper with reviewing committee

(C6) Wang, W., Fam, R., Bao, F., Lepage, Y., and Gao, G. (2019). Neural morphological segmentation model for Mongolian. In *2019 International Joint Conference on Neural Networks (IJCNN–2019)*, pages 1–7, Budapest, Hungary

[1]Numbering follows the document *04-Research achievements publications* submitted together for the degree application.

4.3 Introduction and background

In this section, we describe the morphological analysis task which consists of lemmatisation and MSD analysis subtasks. We introduce the application of the novel concept of analogical grids as a holistic approach to the morphological analysis task.

4.3.1 Morphological analysis task

We address the problem of morphological analysis task:

> Given a **word form**, generate the **lemma** (e.g. the dictionary form of a word) and the **morphosyntactic descriptions (MSD)** of the word form.

In other words, the morphological analysis task consists of two main subtasks, namely lemmatisation and MSD analysis. The MSD is composed of MSFs that characterize the given word form. The number and variety of MSDs may differ between languages, depending on their morphological complexity. Morphologically richer languages tend to have a correspondingly larger number of MSDs. To illustrate the morphological analysis task, Figure 4.1 gives an example in English.

input		output	
		Lemma:	*to analyse*
		MSD:	`Category = verb`
Word form:	*analyses*		`Person = 3`
			`Number = singular`
			`Tense = present`

Figure 4.1: An example of morphological analysis task in English: given the word form *analyses*, generate the lemma *to analyse* and the MSD of the given word. Actual data is one-line tabulation separated text.

4.3.2 Leveraging analogical grids for lemmatisation and MSD analysis

The morphological analysis task is the reciprocal task of morphological generation. It consists of two main subtasks, lemmatisation and MSD analysis. Our proposed method consists of lemmatising inflected forms by solving

play	:	*plays*	:	*playing*
talk	:	*talks*	:	*talking*
treat	:		:	*treating*
analyse	:	*analyses*	:	
read	:	*reads*	:	

puquy	:	*puquyninchikninka*	:	*puquyniykumanta*
qhapaq	:	*qhapaqninchikninka*	:	*qhapaqniykumanta*
intichaw	:		:	*intichawniykumanta*

Figure 4.2: Analogical grids in English (left) and Quechua (right).

analogical equations between the given inflected forms and word forms contained in analogical grids automatically built from the dataset. Similar to the morphological generation task, the analogical grids are built from words represented as vectors with characters and MSFs as features. Candidates are ranked using heuristic features, such as the longest common suffix, the longest common prefix, edit distance, etc. MSD analysis is performed in the same manner by relying on morphological features instead. In summary, we leverage the use of the novel concept of analogical grids in the opposite direction of the morphological generation task.

Let us remember the notion of analogical grids introduced in Chapter 2. Figure 4.2 illustrates two examples of analogical grids, one in English and the other in Quechua. They consist of cells that either contain a word form or are empty. A column or row in an analogical grid usually exhibits similar word forms for different words, such as the infinitive, present 3rd person singular, and present participle for different English verbs. Thus, one can exploit the morphological structure organised in analogical grids to perform morphological analysis by relying on analogical relation between word forms.

4.3.3 Contributions of the chapter

In this chapter, we present several contributions to the field of universal morphological analysis.

- We propose a holistic approach based on the concept of analogical grids. This approach aims to address the universal morphological analysis task by considering the entire word as a unit for analysis;

- We conduct a comprehensive investigation on the performance of morpheme-based, holistic, and neural approaches in over 100 languages with varying degrees of morphological complexity;

- We analyse the impact of the size of training data on the performance improvement of each approach.

4.4 Languages and data used

Experiments were conducted using the SIGMORPHON 2018 Shared Task: Morphological Reinflection Task dataset, originally created for the inflection task. In this study, we inverted the task to morphological analysis. For the details about the dataset, please refer to Section 3.5.

4.5 Experiments

We perform experiments to compare three different approaches: morpheme-based approach, holistic approach, and neural approach. The three approaches are trained on *train* dataset and then tested against the *test* dataset. For the neural approach, the *dev* dataset is used as a validation set during training. Because there is no use of the *dev* dataset for the morpheme-based and holistic approaches, this can be considered a handicap. This issue is discussed in Section 4.7.3.

4.5.1 Morpheme-based approach: decomposing word form into prefix, stem, and suffix

The morpheme-based approach is the same baseline system described in Section 3.7.1 for the morphological generation task. We modify the system to perform the reverse task: morphological analysis. Every instance of training data is analysed using the Levenshtein distance to align the word form and the lemma. Words are broken down into three parts: prefix, stem, and suffix. These affixing rules are grouped based on the given MSD. The difference between the two systems is the part of the analysis of word form in comparison to the generation of inflected form.

In the analysis step, the morpheme-based approach uses the longest common suffixing and prefixing rules to filter the candidates. First, the most frequent and longest common suffixing rule is applied to replace the ending part of the string. In succession to that, the most frequent prefixing rule is applied to generate the predicted lemma. If the system gets an empty predicted lemma, the system will give the form back as its answer. As for the MSD, the system remembers which MSDs correspond to the prefixing and suffixing rule used to produce the predicted lemma. Thus, the highest number of MSDs predicted by the system will be two, one MSD from the prefixing rule and one MSD from the suffixing rule.

Training data

Lemma	Target form	Target MSD
age	*ages*	`V;3;SG;PRS`
age	*aged*	`V;PST`
watch	*watches*	`V;3;SG;PRS`
watch	*watched*	`V;PST`
revise	*revises*	`V;3;SG;PRS`
⋮	⋮	⋮

Question

Form: *analyses*

Answer

Lemma: *analyse*
MSD: `V;3;SG;PRS`

Figure 4.3: An example of given training data and question in English. We are asked to give the lemma: *analyse* and MSD: `V;3;SG;PRS` (third singular present verb) of the English word form: *analyses*.

4.5.2 Holistic approach: analogical grids

In contrast to the morpheme-based approach that involves breaking words into smaller units, we adopt a holistic approach in this work (Singh, 2000; Singh and Ford, 2000; Neuvel and Singh, 2001). Our method involves generating the lemma and its corresponding morphosyntactic description (MSD) by solving analogical equations, which are derived from the patterns observed in the training data. Previous works, like (Marquer et al., 2022), solve morphological analogies through retrieval, while Chan et al. (2022) solve the problem through generation. The holistic approach allows us to capture the overall structure and morphology of the word, rather than just its constituent morphemes.

4.5.2.1 Lemmatisation and MSD analysis by analogy

Let us consider the case where a set of training data (left) and a question (right) are given, as illustrated in Figure 4.3. Initially, we extract all of the analogical grids from the training data. To capture both the form and morphology of the words, we take into account the characters and MSDs as features for the word vector representation. Subsequently, we choose the relevant analogical grid based on the target MSD provided. In the case where multiple analogical equations are possible, we apply heuristic features to select the most suitable candidate.

Figure 4.4 gives an illustration of the generation of the lemma for the

LEMMA	:	V;3;SG;PRS	:	V;PST
age	:	*ages*	:	*aged*
revise	:	*revises*	:	*revised*
compare	:	*compares*	:	*compared*
bake	:	*bakes*	:	

$$ages : age :: analyses : x \Rightarrow x = \boldsymbol{analyse}$$

LEMMA	:	V;3;SG;PRS	:	V;PST
watch	:	*watches*	:	*watched*
miss	:		:	*missed*
publish	:	*publishes*	:	*published*
fetch	:		:	*fetched*

$$watches : watch :: analyses : x \Rightarrow x = \cancel{analys}$$

Figure 4.4: How to generate lemma given the word form (3rd person singular present) *analyses* by solving analogical equations. Different analogical grids may generate different lemmata. The analogical grid on the top produces *analyse*, while the analogical grid on the bottom produces *analys*.

question given in Figure 4.1. According to the given MSD, let us say that there are two analogical grids (top and bottom of the left part of Figure 4.4) extracted from the training data. We construct the following analogical equation:

$$\text{form}_t : \text{lemma}_t :: analyses : \text{lemma}_q$$

given by the first and second columns of the analogical grids according to the given MSD. Here, we rely on one of the heuristic features to give the final answer. Based on the longest common suffix, *analyse* generated by the top analogical grid is selected instead of *analys* which is generated by the bottom one.

$$revises : revise :: analyses : x \quad \Rightarrow x = \boldsymbol{analyse}$$
$$publishes : publish :: analyses : x \quad \Rightarrow x = \cancel{analys}$$

4.5.2.2 Heuristics: selection of candidates

As mentioned previously, there may exist several analogical equations to choose from. We rely on several heuristic features to select one of the candidates as the output:

- edit distance,
- longest common suffix,
- longest common prefix, and
- longest common subsequence.

These heuristics are calculated on the given lemma against lemmata contained in the training dataset.

Similar to the morpheme-based approach, we scanned through the training data to decide whether a language is biased toward a particular affixing phenomenon. Here, we also consider infixing in addition to prefixing and suffixing. This information will decide the feature's precedence to rank the analogical equations For example, if a language is considered biased toward suffixing, the ranking will give priority to the longest common suffix feature; if a language is biased toward infixing, the longest common subsequence feature will be prioritised; and so on.

In the case where the use of heuristic features yields multiple candidate lemmata and MSDs, we solve the analogical equations to generate all possible lemmata and MSDs. The final answer is based on the highest frequency.

4.5.3 Neural approach: sequence-to-sequence network with attention

We consider the morphological analysis task as a sequence-to-sequence problem. Thus, we treat the morphological analysis task as the problem of translating a given word form into its lemma and MSD. The architecture of the model is the same as the one used for the morphological generation task described in Section 3.7.3.

There are two approaches that are used in this experiment: simultaneous (Neural-`sml`) and focus (Neural-`fcs`). The first one performs both subtasks, lemmatisation and MSD analysis, at the same time. The latter approach consists of two models, focusing on one subtask at a time. For both architectures, we use the same hyperparameters defined in Section 3.7.3.2.

4.5.3.1 Neural-sml: simultaneously perform lemmatisation and MSD analysis

We assume that the *end-to-end* paradigm is the simplest neural approach for morphological analysis. This neural approach performs both the lemmatisation and MSD analysis subtasks at the same time. The input to the network is the sequence of characters appearing in the word form. We train separate models for each language and their respective training data sizes.

$$\mathrm{f_1} \quad \mathrm{f_2} \quad \ldots \quad \mathrm{f_n}$$

The output of the network is the sequence of MSFs followed by the sequence of characters of the lemma. A special token, =|=, is used to separate the MSD and lemma.

$$\mathrm{MSF_1} \quad \mathrm{MSF_2} \quad \ldots \quad \mathrm{MSF_i} \quad =|= \quad \mathrm{l_1} \quad \mathrm{l_2} \quad \ldots \quad \mathrm{l_j}$$

4.5.3.2 Neural-fcs: focus on one subtask at a time

In contrast to the previous neural approach which performs the two subtasks at the same time, this neural approach consists of two models for each language and training data size. One model focuses on learning to output the sequence of characters of the lemma. The other one handles predicting which MSFs are related to the given word form. This means that for a language with three training data sizes (*low*, *medium* and *high*), we have six different models.

The input of both models is the same, the sequence of characters of the word form. The output is different according to the subtask it handles: lemmatisation or MSD analysis. The model for lemmatisation will output only the lemma of the given word form. The model for MSD analysis will output the sequence of MSFs.

4.5.4 Evaluation metrics

The performance of the systems is evaluated on both subtasks, lemmatisation and MSD analysis. The lemma is evaluated as a string, while the MSD is evaluated as a set.

4.5.4.1 Lemmatisation

For the lemmatisation subtask, we use **accuracy** and average Levenshtein distance to evaluate the performance of the systems. These are the same metrics used in the morphological generation task. Please refer to Section 3.7.5 for the exact definitions.

4.5.4.2 MSD analysis

For the MSD analysis task, we use **precision** and **recall** defined below to measure the performance of the systems. In addition, we derive the **F1 score** which is the harmonic mean of the precision and recall. TP is the number of true positive samples, FP is the number of false positive samples and FN is the number of false negative samples.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (4.1) \qquad \text{Recall} = \frac{TP}{TP + FN} \quad (4.2)$$

$$\text{F1} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.3}$$

4.6 Results and analysis

For each system, we evaluate the predicted lemma and MSD. Table 4.1 shows the overall results for the lemmatisation subtask, while Table 4.2 presents the results for the MSD analysis subtask. Please refer to Tables D.3, D.4, D.5, D.6 and D.7 in Appendix D for more detailed results in each language.

4.6.1 Morphological analysis: lemmatisation

Table 4.1 shows the overall accuracy and average Levenshtein distance of the performance of the systems on all 103 languages and all data sizes for the lemma. The morpheme-based approach has a slight lead in accuracy, less than 1%, on the *low* dataset. However, the holistic approach performs better on the *medium* and the *high* datasets. The neural approaches perform poorly on the *low* dataset. Neural-`sml` performs the worst, while Neural-`fcs` achieves a better accuracy by +10%. However, both of them are far behind morpheme-based and holistic approaches. The differences are less than 30% for Neural-`sml` and 20% for Neural-`fcs`. Neural approaches start to outperform the morpheme-based approach and the holistic approach on the *medium* dataset.

For the average Levenshtein distance, we can observe that the morpheme-based approach consistently leads by a small margin on all three datasets in comparison to the holistic approach. This tells us that the morpheme-based approach is usually pretty close to the right answers, although, the holistic approach is not that far behind. For neural approaches, a similar analysis for accuracy can be made. Both neural approaches suffer on the *low* dataset and start to overcome the other systems when the size of the dataset increases.

Table 4.1: Average accuracy and Levenshtein distance on *test* dataset for lemmatisation subtask.

Method	**Accuracy**			**Levenshtein distance**		
	low	**medium**	**high**	**low**	**medium**	**high**
Morpheme	43.6	60.9	71.0	1.3	0.9	0.7
Holistic	42.8	63.1	73.7	1.7	1.0	0.8
Neural-`sml`	15.3	75.3	89.1	3.8	0.5	0.2
Neural-`fcs`	26.7	80.8	91.6	2.8	0.4	0.2

Table 4.2: Average precision, recall and F1 score on *test* dataset for MSD analysis subtask.

Method	**Precision**			**Recall**			**F1 score**		
	low	**medium**	**high**	**low**	**medium**	**high**	**low**	**medium**	**high**
Morpheme	0.72	0.79	0.82	0.72	0.80	0.82	0.72	0.79	0.82
Holistic	0.67	0.76	0.79	0.84	0.88	0.87	0.75	0.82	0.83
Neural-`sml`	0.59	0.83	0.88	0.59	0.83	0.88	0.59	0.83	0.88
Neural-`fcs`	0.63	0.84	0.88	0.63	0.84	0.88	0.63	0.84	0.88

4.6.2 Morphological analysis: MSD analysis

Let us now turn to the evaluation of MSD analysis. Table 4.2 presents the average precision, recall and F1 score of the systems over all the languages.

The morpheme-based approach has higher precision, while the holistic approach has a higher recall. This is due to how the two approaches work in different ways. The morpheme-based approach will only output no more than two MSD candidates, one from the prefixing rule and another one from the suffixing rule. This leads to a precision-focused performance.

As for the holistic approach, the system will have more candidates to consider due to its flexibility by treating the whole word as a unit. It is not bounded by the affixing rule (for example, prefixes and suffixes). By having more candidates, the performance of the system is biased towards recall. However, the trade-off on precision and recall between the morpheme-based and holistic approaches is summarised by the F1 score. Our holistic approach has a higher F1 score in comparison to the morpheme-based approach in all different training data sizes. On top of that, our holistic approach achieves the best F1 score under the *low* data conditions.

Neural approaches also perform poorly under the *low* data conditions at the MSD analysis subtask. They begin to perform better than the other approaches when the training data size increases. We also find that similar to the lemmatisation subtask, the neural-`fcs` performs better than neural-`sml`. Our hypothesis is that the neural approach will encounter fewer exceptions to learn and have a higher generalisation power by focusing on one task at a time.

4.7 Discussion and further experiments

In this section, we discuss again the analysis of the results we obtained from previous experiments. Furthermore, we provide a discussion on how the performance may be affected by the additional data from the validation dataset. By nature, the morpheme-based and holistic approaches are not using this resource. This can be seen as a handicap relative to the neural approach. Another thing that is interesting to look at further is the performance's turning point. In the following section, we perform more experiments to inspect the performance curve by cutting the size of the data into finer granularity.

4.7.1 Morpheme vs. holistic: accuracy vs. exactness

Based on the experimental results, we observe that the holistic approach predicted more exact lemmata. On the other hand, the morpheme-based approach produced lemmata closer to the answers when it produced the wrong answers.

From the MSD analysis experiments, we understand that the morpheme-based approach is more precise when predicting MSDs, while the holistic approach achieved higher recall. However, our holistic approach achieves higher F1 score in comparison to the morpheme-based approach.

4.7.2 Trade-off between low and high-resource conditions

Looking at the performance of the systems on different training data sizes, it is clear that neural approaches suffer under low-resource conditions. This is common knowledge. The morpheme-based and holistic approaches are better under low-resource conditions even in the absence of the *dev* dataset. The morpheme-based and holistic approaches are around 2 to 3 times better than the neural approaches in the *low* training data size. Neural approaches

Table 4.3: Average accuracy and Levenshtein distance scores on *test* dataset for lemmatisation subtask with and without the help of *dev* dataset.

Method	**Accuracy**			**Levenshtein distance**		
	low	**medium**	**high**	**low**	**medium**	**high**
Morpheme	43.6	60.9	71.0	1.3	0.9	0.7
Morpheme+`dev`	55.6	62.9	71.3	1.0	0.8	0.7
Holistic	42.8	63.1	73.7	1.7	1.0	0.8
Holistic+`dev`	56.9	65.4	74.0	1.2	0.9	0.8

achieve the best performance in comparison to the other approaches when the size of the training data increases.

4.7.3 Improvement on using additional data from validation dataset

As mentioned in Section 4.5, unlike the neural approaches which use the *dev* dataset as a validation dataset in the training phase, both morpheme-based and holistic approaches do not use the *dev* dataset at all. It can be seen as an advantage for the neural approach. We perform further experiments to analyse the impact of having more training data for the morpheme-based and holistic approaches. In this case, we consider using the *dev* as additional training data. By having the *same* amount of data, we would like to investigate whether the morpheme-based and holistic approaches are able to improve their performance or not. The next question is whether the morpheme-based and holistic approaches are able to catch up with the neural approach or not.

Table 4.3 shows the comparison of the system's performance on lemmatisation subtask with and without the help of the *dev* dataset as training data. We observe that the holistic approach gains an improvement of up to 14% on the *low* dataset. The improvement slows down towards the *high* dataset. This is also true for the average Levenshtein distance.

The same improvement can also be observed in the MSD analysis subtask. Table 4.4 presents the comparison of the performance of the system with and without the use of the *dev* dataset while training. For both the morpheme-based and holistic approaches, there is an improvement in F1 score on the *low* dataset, 0.06 for the morpheme-based approach and 0.05 for the holistic approach. However, there is no improvement in the *medium* and the *high* datasets for the holistic approach. We also observed that the trade-off be-

Table 4.4: Average precision, recall and F1 score on *test* dataset for MSD analysis subtask with and without the help of *dev* dataset.

Method	**Precision**			**Recall**			**F1 score**		
	low	**medium**	**high**	**low**	**medium**	**high**	**low**	**medium**	**high**
Morpheme	0.72	0.79	0.82	0.72	0.80	0.82	0.72	0.79	0.82
Morpheme+`dev`	0.78	0.80	0.82	0.78	0.80	0.82	0.78	0.80	0.82
Holistic	0.67	0.76	0.79	0.84	0.88	0.87	0.75	0.82	0.83
Holistic+`dev`	0.74	0.77	0.79	0.88	0.88	0.87	0.80	0.82	0.83

tween precision and recall of the two systems still leads to a higher F1 score for our holistic approach.

Through this experiment, we observe that the additional training data (in this case, *dev* dataset), helps improve the performance of morpheme-based and holistic approaches, especially in the *low* dataset. Both systems leave the neural approach further behind, with around 40% to Neural-`sml` and 30% to Neural-`fcs` for the lemmatisation subtask. The same thing can be observed for the MSD analysis subtask. Morpheme+`dev` and Holistic+`dev` approaches have higher F1 scores, almost 0.2 points. On top of it, the Holistic+`dev` leads the performance of lemmatisation with more than 1% accuracy and has a 0.02 higher F1 score in comparison to the morpheme-based approach.

4.7.4 Looking deeper on the performance curve

To understand the turning point between different systems, we carry out further experiments on different sizes of training data sizes. This is done to have more points on the performance curve of the systems.

Figure 4.5 shows the graph of the performance of the systems for the accuracy of the lemmatisation subtask. Both morpheme-based and holistic approaches have a relatively flat curve from *low* to *high* dataset. In contrast, neural approaches: Neural-`sml` and Neural-`fcs` have a steep curve from *low* (100) to *medium* (1k) dataset. It is then becoming flat towards the *high* (10k) dataset. We observe that the neural approaches start to perform better than morpheme-based and holistic approaches when the training data size is around 500 to 750.

Figure 4.6 shows the graph of the performance of the systems for the F1 score on the MSD analysis subtask. Due to very similar results, we decided to zoom the y-axis to allow better observation. Similar to the performance of the lemmatisation subtask, the neural approaches start to perform better

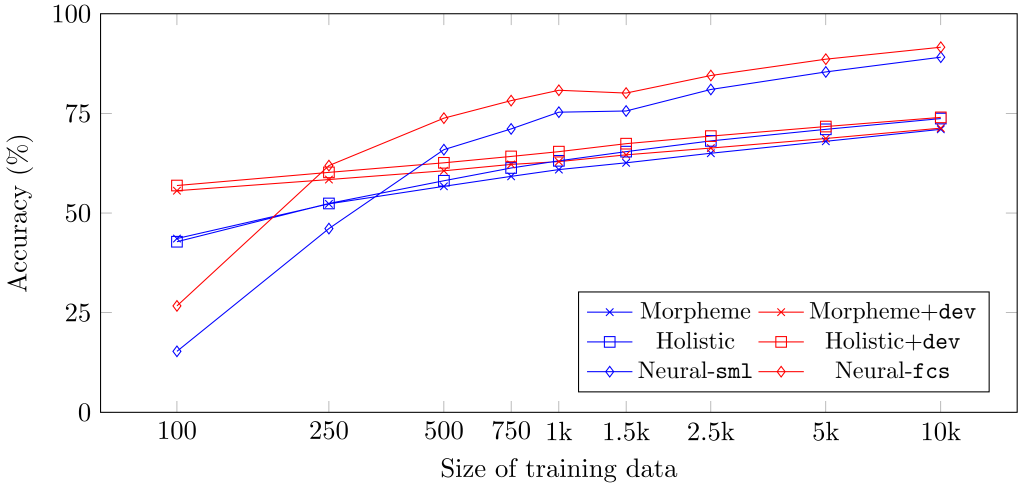


Figure 4.5: Accuracy of systems trained on different sizes of dataset for lemmatisation subtask. Caution: log scale on the abscissae.

on training data sizes of around 500 to 750. As also observed in the previous figure, neural approaches start with a very low F1 score in comparison to morpheme-based and holistic approaches. Neural approaches perform better on the bigger dataset. We can also see that Neural-`sml` which has the lower F1 score on the smaller dataset is able to close the gap with Neural-`fcs` at 10k. Here, we find that there is an improvement in morpheme-based and holistic approaches from 100 to 500. The curve seems to reach a plateau afterwards. However, Morpheme+`dev` and Holistic+`dev` have a very flat curve even from training data size of 100.

4.8 Summary of the chapter

We developed several systems to perform universal morphological analysis. Experiments were carried out with the SIGMORPHON 2018 Shared Task dataset which is used in the experiments described in Chapter 3. It consists of 103 languages from various language families with various morphological richness. The performance of the systems is evaluated on both, the lemmatisation and MSD analysis, separately. Since this task does not exist in the SIGMORPHON campaign, there is no system from outside to be compared with. The holistic approach predicted more accurate lemmas, while the morpheme-based approach produced closer lemmas to the answers according

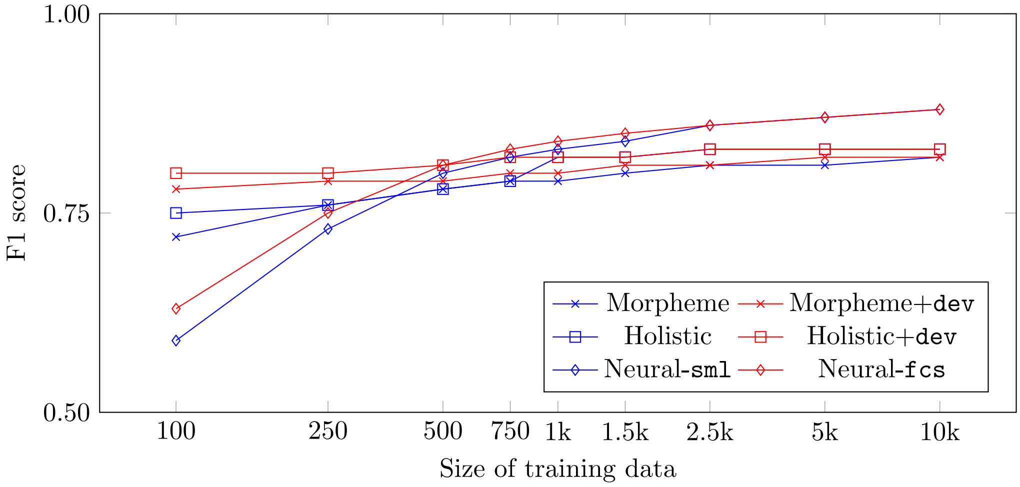


Figure 4.6: F1 score of systems trained on different sizes of dataset for MSD analysis subtask. Please notice that the y-axis is zoomed to a range of [0.50, 1.00] to give a better view due to very similar results between systems. Caution: log scale on the abscissae.

to Levensthein edit distance. For the MSD analysis subtask, the morpheme-based approach is more precise, while the holistic approach achieves higher recall. However, we found that the trade-off between precision and recall of the two systems leads to the holistic approach having a higher F1 score. Neural approaches performed the worst under *low* data conditions but started to overcome the other approaches when there was enough data to train.

Based on these results, we summarise our conclusions as follows.

- The holistic and morpheme-based approaches perform better in comparison to the neural approach under low-resource scenarios. Our holistic approach outperforms the other approaches when it is allowed to use of *dev* dataset.

- For the lemmatisation task, the holistic approach performs similarly or slightly better than the morpheme-based approach in accuracy.

- For the MSD analysis task, the morpheme-based approach favours precision while the holistic approach favours recall. The holistic approach achieves a higher F1 score.

- Two neural approaches trained on the two specific subtasks perform better than one trained on the two subtasks at the same time.

Chapter 5

Conclusion and future works

This chapter summarises this thesis. It presents the conclusion based on the results obtained in the previous chapters and contributions made by this thesis. At last, it provides future direction for research in both the concept of analogical grids and its application to morphological tasks.

5.1 Conclusion

We proposed a pipeline for the production of analogical grids from words contained in a given corpus by relying on a formalization of analogy. The analogical grids are built in an agnostic way, without any a priori linguistic knowledge, relying on the sole form of the words, without decomposing them into components and without taking any frequency information into account.

Without surprise, languages known to be richer in morphology produce bigger and more analogical grids than languages less rich in morphology. Empty cells in such analogical grids are interesting because they could be filled by words that should then be tested against the actual language. We observed an interesting phenomenon when producing analogical grids in four different languages on translations of the same text. It relates the saturation of the obtained analogical grids to their size. Experimental results show that the coefficients which characterize the relation would not be influenced by the size, the genre or the language of texts.

We carried out experiments to see how many of the words used by an author can be predicted from such analogical grids in comparison to another author. The results obtained in a variety of languages of the world, with two different thresholds for the density of the analogical grids produced, can be used to characterize the relative morphological richness of languages as well as the richness of the vocabulary of authors.

Further experiments are conducted in Indonesian to confirm the explanation of the unseen words. We first explain the unseen words on the level of surface form by extracting all possible analogical clusters from the words contained in the training set which included the unseen words. The explanations are then confirmed on two additional levels of representation: morphological and distributional semantic representation. Results from ten-fold cross-validation show that more than 98% of the unseen words can be explained on the level of surface form. The remaining unseen words are mostly: plurals (formed by repetition in Indonesian, which is excluded in our formalisation) and proper nouns. As a final result, almost half of the unseen words can be explained on three different levels: surface form, morphology, and distributional semantics at the same time.

Leveraging the novel concept of analogical grids, we developed a holistic method for the morphological generation task. Morphological generation task is a morphological task given a lemma and the target MSD, generating its inflected form. This task is a standard task in the yearly evaluation campaign SIGMORPHON Shared Task: Morphological Reinflection Task. This thesis is aligned with the current research direction and the main subject in the morphological reinflection area. Systems developed in this campaign are released publicly as available language tools. Experiments are carried out on the 2018 Shared Task which offers the largest number of languages. This allows us to evaluate the performance across many languages and against publicly available tools. We proposed a holistic approach to the problem of morphological generation by treating the word form as a unit instead of breaking down word forms into smaller pieces, like morphemes, as is done in some baseline systems. The structural information and rich morphological features of word forms are used to build feature vector representations. Reinflected forms are generated by solving analogical equations between word forms encapsulated in analogical grids. We evaluate the performance of three approaches: morpheme-based (baseline system), holistic, and neural approaches. Experimental results show that our proposed holistic approach improves the accuracy of morphological generation by 1.3% (up to 20% when allowed to use the validation dataset) in low-resourced conditions (100 training instances only) and 0.1 in edit distance under high-resource conditions. Our proposed method outperforms neural approaches under high-resource conditions (10,000 training instances) for languages like Albanian, Czech, Haida, etc.

We also applied the novel concept of analogical grids to the morphological analysis task. This task is the reciprocal task of morphological generation. It consists of two main subtasks, lemmatisation and MSD analysis. Our proposed method consists of lemmatizing inflected forms by solving analogical equations between the given inflected forms and word forms contained in analogical grids automatically built from the dataset. Candidates are ranked using heuristic features, such as the longest common suffix, the longest common prefix, edit distance, etc. MSD analysis is performed in the same manner by relying on morphological features instead. We compare the performance of morpheme-based, holistic, and neural approaches. For the lemmatisation subtask, experimental results show that the holistic approach predicted almost 3% (up to 13% when allowed to use the validation dataset) more accurate lemmata with a 0.2 better score in edit distance under low-resource conditions (100 training instances). For the MSD analysis subtask, the holistic approach achieves a 0.02 better F1 score in comparison to the morpheme-based approach. Although neural approaches achieve the best performance under high-resource conditions, they suffer under low-resource conditions and perform the worst in comparison to the other two proposed approaches. In summary, our holistic approach leads under low-resource conditions.

As a summary, we addressed the issue of explaining unseen words by using computational analogy. We proposed a novel concept called analogical grid which captures the organisation of the lexicon in a language. The phenomenon observed regarding the saturation and size of analogical grids may relate to the confidence in filling empty cells in it. We showed how to use them to explain and generate unseen words by leveraging the concept of analogical grids and applying it to morphological tasks: morphological generation and morphological analysis. On average, our proposed holistic approach, which leverages the novel concept of analogical grids, performed better than the morpheme-based approach on both the morphological generation and analysis tasks. Under low-resource conditions, our proposed holistic approach with the proposed concept of analogical grids outperforms morpheme-based and neural approaches in accuracy for morphological generation, and in accuracy, precision, and recall for morphological analysis. In addition, it outperformed the winner system of the 2018 edition of the SIGMORPHON Shared Task on morphological generation task. Furthermore, our proposed holistic approach is a lazy learning method which results in a more efficient approach towards:

- **time**: no need for the training phase, and
- **storage**: no trained model to be saved.

To be fair, the efficiency mentioned above should be multiplied by a factor of the total number of languages. For example, neural approaches require training a model for each language as it is a language-dependent approach. In contrast to that, our proposed method is a language-independent approach.

5.2 Future work

In this section, we discuss directions that can be considered to be done in the future. We divide the future works into two main parts:

- Semantical analogical grids: Induction on the level of semantics
- Improvements in morphological tasks

Let us remember that there are three levels of induction: surface form, morphological, and semantics. In this work, we focus on the first two levels because we want to handle is morphological task. We might also want to consider the level of semantics to construct different kinds of analogical grids.

In this case, one may consider using word embeddings as a popular distributional semantics representation to produce semantical analogical grids. As introduced in Chapter 2, the algorithm presented in this thesis is limited to using only an integer as the value of the vector's dimension. (Shu and Nakayama, 2017) shows how to compose integer codes from floating numbers. In this way, we may use the compositional codes as the input of our algorithm as illustrated in Figure 5.1.

$$\textit{jakarta} = \begin{pmatrix} -0.0035 \\ 0.0497 \\ -0.6938 \\ -0.0237 \\ \vdots \\ 0.3603 \end{pmatrix} \rightarrow \begin{pmatrix} 110 \\ 23 \\ 236 \end{pmatrix} \qquad \begin{array}{ccccc} \textit{japan} & : & \textit{tokyo} & : & \textit{japanese} \\ \textit{indonesia} & : & \textit{jakarta} & : & \textit{indonesian} \\ \textit{france} & : & & : & \textit{french} \\ \textit{germany} & : & \textit{berlin} & : & \end{array}$$

Figure 5.1: Extracting compositional codes (left) from word embeddings to construct semantical analogical grid (right)

Another way is to have seed clusters in advance. We extend a seed cluster with more pairs of strings contained in the vocabulary according to a threshold. The use of a threshold allows us to have a more relaxed constraint while dealing with floating numbers.

Particularly for the morphological generation task, experimental results show that our proposed method performed better than the baseline. However, according to oracle experiments, there is still some room for improvements, especially to handle unseen morphological features list.

The use of a subword regularisation algorithm to have a middle approach between morpheme and holistic approach. It will be similar to a morpheme-based approach by capturing *pseudo-affix* phenomena.

To handle the high complexity of the language, we may consider using Principal Component Analysis (PCA). The idea is to reduce the complexity of the knowledge that needs to be learned during the training phase by having a more compact representation of the data. We hope that having more compact data will lead to a smaller model which will also help to improve the generalisation power of the model.

Appendix A

Pledge for Thesis Submission

I (FAM Rashel Putraruddy Scala) hereby confirm that:

(1) this thesis *'Analogical grids: study on morphological reinflection, lemmatisation and morphosyntactic description analysis'* submitted in partial fulfilment for the degree of Doctor (Engineering) at the Graduate School of Information, Production and Systems, Waseda University, is my original work.

(2) I have upheld the principles of academic integrity, and I certify that:

- there is no data falsification in this thesis,
- there is no data fabrication in this thesis,
- there is no plagiarism in this thesis.

(3) this thesis has not been submitted previously or concurrently and, will not be submitted by myself in the future, for any other degree at any other institution.

I am fully aware that if I should violate any of the above commitments, I will be subject to strict disciplinary action (indefinite suspension from the University, invalidation of grades for the semester, failure to pass the master's thesis, etc.) and my degree will be revoked even after I have received my degree.

Name: FAM Rashel Putraruddy Scala Student N^{o}: 44172512

Signature: Date: September 4th, 2023

Appendix B

Publications

B.1 Journals

(J1) Fam, R. and Lepage, Y. (2022). Organising lexica into analogical grids: A study of a holistic approach for morphological generation under various sizes of data in various languages. *Journal of Experimental & Theoretical Artificial Intelligence*, 0(0):1–26

(J2) Fam, R. and Lepage, Y. (2021). A study of analogical density in various corpora at various granularity. *Information*, 12(8)

B.2 International conferences with reviewing committee

(C1) Fam, R. and Lepage, Y. (2023a). Investigating parallelograms: Assessing several word embedding spaces against various analogy test sets in several languages using approximation. In *Proceedings of the 10th Language and Technology Conference (LTC–2023)*, pages 68–72, Poznań, Poland. Fundacja uniwersytetu im. Adama Mickiewicza

(C2) Lo, H.-W., Yifei, Z., Fam, R., and Lepage, Y. (2022). A study of regenerating sentences given similar sentences that cover them on the level of form and meaning. In *Proceedings of the 36th Pacific Asia

Numbering follows the document *04-Research achievements publications* submitted together for the degree application.

Conference on Language, Information and Computation (PACLIC-36), pages 369–378, Manila, Philippines. De La Salle University

(C3) Yifei, Z., Fam, R., and Lepage, Y. (2022). Extraction of analogies between sentences on the level of syntax using parse trees. In *Proceedings of the workshop Analogies: from Theory to Applications (ATA@ICCBR 2022), held with the 30th International Conference on Case-Based Reasoning*, pages 1 – 13, Nancy, France

(C4) Putro, S. C., Jiono, M., Nuraini, N. P., and Fam, R. (2021). Development of statistics teaching materials using augmented reality to reduce misconception. In *2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE)*, pages 151–156

(C5) Fam, R. and Lepage, Y. (2019). A study of analogical grids extracted using feature vectors on varying vocabulary sizes in Indonesian. In *Proceedings of 2019 International Conference on Advanced Computer Science and Information Systems (ICACSIS–19)*, pages 255–260, Bali, Indonesia

(C6) Wang, W., Fam, R., Bao, F., Lepage, Y., and Gao, G. (2019). Neural morphological segmentation model for Mongolian. In *2019 International Joint Conference on Neural Networks (IJCNN–2019)*, pages 1–7, Budapest, Hungary

(C7) Fam, R. and Lepage, Y. (2018a). IPS-WASEDA system at CoNLL–SIGMORPHON 2018 shared task on morphological inflection. In *Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection (CoNLL–18)*, pages 33–42, Brussels. Association for Computational Linguistics

(C8) Fam, R. and Lepage, Y. (2018b). Tools for The Production of Analogical Grids and a Resource of N-gram Analogical Grids in 11 Languages. In chair), N. C. C., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Hasida, K., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S., and Tokunaga, T., editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC–2018)*, Miyazaki, Japan. European Language Resources Association (ELRA)

(C9) Fam, R., Purwarianti, A., and Lepage, Y. (2018). Plausibility of word forms generated from analogical grids in Indonesian. In *Proceedings of*

the 16th International Conference on Computer Applications (ICCA–2018), pages 179–184, Yangon, Myanmar. UCSY

(C10) Fam, R. and Lepage, Y. (2017a). A holistic approach at a morphological inflection task. In *Proceedings of the 8th Language and Technology Conference (LTC–2017)*, pages 88–92, Poznań, Poland. Fundacja uniwersytetu im. Adama Mickiewicza

(C11) Fam, R. and Lepage, Y. (2017b). A study of the saturation of analogical grids agnostically extracted from texts. In *Proceedings of the Computational Analogy Workshop at the 25th International Conference on Case-Based Reasoning (ICCBR-CA–2017)*, pages 11–20, Trondheim, Norway

(C12) Fam, R., Lepage, Y., Gojali, S., and Purwarianti, A. (2017b). A study of explaining unseen words in Indonesian using analogical clusters. In *Proceedings of the 15th International Conference on Computer Applications (ICCA–2017)*, pages 416–421, Yangon, Myanmar

(C13) Fam, R. and Lepage, Y. (2016b). Morphological predictability of unseen words using computational analogy. In *Proceedings of the Computational Analogy Workshop at the 24th International Conference on Case-Based Reasoning (ICCBR-CA–2016)*, pages 51–60, Atlanta, Georgia

(C14) Rashel, F., Luthfi, A., Dinakaramani, A., and Manurung, R. (2014). Building an Indonesian rule-based part-of-speech tagger. In *Proceedings of 2014 International Conference on Asian Language Processing (IALP–2014)*, pages 70–73, Kuching, Malaysia

(C15) Dinakaramani, A., Rashel, F., Luthfi, A., and Manurung, R. (2014). Designing an Indonesian part of speech tagset and manually tagged Indonesian corpus. In *Proceedings of 2014 International Conference on Asian Language Processing (IALP–2014)*, pages 66–69, Kuching, Malaysia

(C16) Rashel, F. and Manurung, R. (2014). Pemuisi: a constraint satisfaction-based generator of topical Indonesian poetry. In *Proceedings of the Fifth International Conference on Computational Creativity (ICCC–2014)*, pages 82–90, Ljubljana, Slovenia

(C17) Rashel, F. and Manurung, R. (2013). Poetry generation for Bahasa Indonesia using a constraint satisfaction approach. In *Proceedings of 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS–2013)*, pages 219–224, Bali, Indonesia

B.3 Conferences without reviewing committee

(P1) Fam, R. and Lepage, Y. (2023c). A resource of sentence analogies on the level of form extracted from corpora in various languages. In *Proceedings of the 29th Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2023)*, pages 103–107, Okinawa, Japan

(P2) Fam, R. and Lepage, Y. (2023b). Investigating parallelograms inside word embedding space using various analogy test sets in various languages. In *Proceedings of the 29th Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2023)*, pages 718–722, Okinawa, Japan

(P3) Fam, R., Liu, P., and Lepage, Y. (2019). Checking the validity of word forms generated to fill empty cells in analogical grids. In *Proceedings of the 25th Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2019)*, pages 530–533, Nagoya, Japan

(P4) Fam, R. and Lepage, Y. (2018c). Validating analogically generated Indonesian words using Fisher's exact test. In *Proceedings of the 24rd Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2018)*, pages 312–315, Okayama, Japan

(P5) Fam, R., Purwarianti, A., and Lepage, Y. (2017c). Plausibility of word forms generated from analogical grids in Indonesian. In *11th International collaboration Symposium on Information, Production and Systems (ISIPS–2017)*, pages 245–247, Kitakyushu, Japan

(P6) Fam, R., Lepage, Y., Gojali, S., and Purwarianti, A. (2017a). Indonesian unseen words explained by form, morphology and distributional semantics at the same time. In *Proceedings of the 23rd Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2017)*, pages 178–181, Tsukuba, Japan

(P7) Fam, R. and Lepage, Y. (2016a). An empirical property of the density of paradigm tables. In *10th International collaboration Symposium on Information, Production and Systems (ISIPS–2016)*, page (no pagination), Kitakyushu, Japan

Appendix C

List of algorithms

There are two main algorithms which correspond to the production of analogical grids:

- **Algorithm 1**: extraction of analogical clusters from a given set of words (or even any strings)
- **Algorithm 2**: production of analogical grids from a set of analogical clusters

Algorithm 1 Building a set of analogical clusters from a set of words

function BUILD_CLUSTERS(*set of words*)
  tree $\leftarrow$ from the *set of words* $\triangleright$ Hierarchically group words by their number of occurrences of characters.
  repeat top-down exploration of the *tree* against itself
    group pairs of words by equal difference
      of number of occurrences of characters
  until last character
  for all *set of word pairs* with equal number of occurrences of characters **do**
    CHECK_DISTANCE(*set of word pairs*)
  end for
end function

function CHECK_DISTANCE(*set of word pairs* $(A_1, B_1), \ldots, (A_n, B_n)$)
  for all $i \in \{1, \ldots, n\}$ **do**
    compute $\mathrm{d}(A_i, B_i)$
  end for
  for all *set of word pairs* (A_i, B_i) with same distance **do**
    CHECK_CLUSTER(*set of word pairs*)
  end for
end function

function CHECK_CLUSTER(*set of word pairs* $(A_1, B_1), \ldots, (A_n, B_n)$)
  $\mathcal{V} \leftarrow \{1, \ldots, n\}$ $\triangleright$ Vertices of the graph.
  $\mathcal{E} \leftarrow \{(i, j) \in \mathcal{V}^2 \ / \ A_i : A_j = B_i : B_j\}$ $\triangleright$ Edges of the graph.
  list $\leftarrow$ *nodes* in $\mathcal{V}$ sorted by non-increasing number of edges
  not_yet_covered $\leftarrow \mathcal{V}$
  repeat
    $i \leftarrow$ first node in *list*
    delete i from *list*
    if $i \in$ *not_yet_covered* **then**
      clique $\leftarrow \{i\}$ $\triangleright$ Initialize clique to singleton of not yet explored vertex.
      clique, *not_yet_covered* $\leftarrow$ EXPAND_CLIQUE(*clique*, *list*, *not_yet_covered*)
      return *clique* $\triangleright$ *clique* is an analogical cluster.
    end if
  until *not_yet_covered* $= \emptyset$
end function

```
function EXPAND_CLIQUE(clique, list, not_yet_covered)
    for all i in list do
        if i is connected with all vertices in the clique then
            add i to the clique                         ▷ Remains a clique.
            delete i from not_yet_covered
        end if
    end for
    return clique, not_yet_covered
end function
```

Algorithm 2 Building a set of analogical grids from a set of analogical clusters

```
function BUILD_PARADIGM_TABLES(set of analogical clusters, threshold)
    tables ← ∅                          ▷ Set of paradigm tables, initially empty.
    list ← set of analogical clusters sorted by non-increasing order of size
    repeat
        analogical cluster ← first analogical cluster in list
        delete analogical cluster from list
        table ← analogical cluster  ▷ Make analogical cluster an analogical grid.
                                    ▷ By construction, it has only 2 columns
                                    ▷ and a density of 100 %.
        table, list ← EXPAND_TABLE(table, list, threshold)
        tables ← tables ∪ {table}
    until list is empty
    return tables
end function

function EXPAND_TABLE(table, list, threshold)
    repeat                              ▷ Possibly scan the list several times.
        for all cluster in the list (in non-increasing order of sizes) do
            if cluster can be added to table and density of new table ≥ threshold then
                add cluster to table (either transposed or not)
                delete cluster from list
            end if
        end for
    until no cluster can be added to table
    return table, list
end function
```

Appendix D

List of additional tables

These are list of additional tables:

- **Table D.1**: Accuracy at morphological generation task in each language for baseline system (morpheme-based), holistic approach, our seq2seq model with and without data augmentation.
- **Table D.2**: Comparison of average accuracy and Levenshtein distance scores between our holistic approach and systems submitted to the SIGMORPHON 2018 Shared Task.
- **Table D.3**: Accuracy at morphological analysis task in each language for baseline system (morpheme-based), holistic approach, our neural approaches: neural-`sml` and neural-`fcs`.
- **Table D.4**: Same as previous table but for average Levenshtein distance.
- **Table D.5**: Same as previous table but for precision.
- **Table D.6**: Same as previous table but for recall.
- **Table D.7**: Same as previous table but for F1 score.
- **Table D.8**: Morphological complexity (C_{WALS}) for all dataset sizes computed using the Formula (3.8).
- **Table D.9**: Estimated morphological complexity (C_{WALS}) for all dataset sizes computed using the Formula (3.8).
- **Table D.10**: Number of productive, unproductive, and total rule computed on each language using Tolerance Principle (Formula (3.5)).

Table D.1: Accuracy at morphological generation task in each language for morpheme-based system (**M**), holistic approach(**H**), our neural approach (**S**) and with data augmentation (**S-Aug**).

Language	Accuracy											
	low				medium				high			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
adyghe	59.0	72.1	35.5	73.8	84.8	87.0	88.0	89.5	91.6	91.1	95.6	95.2
albanian	5.8	23.8	0.6	11.6	13.2	72.1	44.8	65.2	12.5	88.3	81.1	80.5
arabic	26.8	27.9	0.1	21.0	39.5	48.6	61.1	67.9	47.0	62.5	93.0	91.7
armenian	37.0	33.2	1.2	34.2	70.4	77.8	76.5	83.7	86.6	88.0	94.1	90.9
asturian	58.6	57.6	19.7	53.1	89.1	88.4	87.4	89.7	95.2	94.4	97.8	97.2
azeri	24.0	26.0	13.0	37.0	50.0	55.0	69.0	67.0	70.0	74.0	81.0	82.0
bashkir	39.4	41.4	11.5	35.9	72.6	72.9	87.0	81.0	90.7	88.8	94.1	92.6
basque	0.1	0.1	1.9	8.6	1.9	2.3	67.0	79.2	7.3	8.0	97.4	96.9
belarusian	6.8	10.6	4.6	5.7	21.5	25.9	44.6	55.4	41.0	38.7	85.3	80.9
bengali	50.0	49.0	14.0	49.0	76.0	74.0	94.0	96.0	81.0	83.0	98.0	99.0
breton	20.0	20.0	18.0	61.0	67.0	72.0	83.0	88.0	73.0	73.0	91.0	92.0
bulgarian	30.7	32.4	4.3	49.8	70.8	74.1	70.6	82.1	89.0	88.9	95.4	94.3
catalan	60.8	57.1	4.6	32.6	85.6	83.9	85.0	92.3	95.7	94.6	98.1	95.9
classical-syriac	94.0	92.0	41.0	72.0	99.0	99.0	94.0	98.0	97.0	96.0	98.0	100.0
cornish	10.0	12.0	7.5	22.5	12.0	8.0	47.5	57.5				
crimean-tatar	56.0	67.0	16.0	63.0	78.0	80.0	95.0	89.0	95.0	93.0	99.0	98.0
czech	38.5	38.5	1.6	26.1	79.9	81.7	51.1	76.6	90.6	90.8	85.5	86.3
danish	58.3	64.9	30.2	53.0	77.8	79.1	74.3	69.8	87.0	86.5	91.3	85.8
estonian	21.5	19.2	0.7	28.4	62.9	61.1	60.0	70.3	78.0	78.0	90.6	88.0

Continued on next page

Table D.1 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
faroese	34.4	39.2	3.3	16.6	65.2	68.1	51.0	60.6	76.1	76.6	79.8	74.5
finnish	10.4	16.1	0.7	18.7	44.1	41.5	42.6	69.9	78.0	76.6	84.1	82.0
friulian	70.0	71.0	25.0	49.0	92.0	92.0	89.0	94.0	96.0	97.0	98.0	99.0
galician	53.0	51.1	9.1	30.7	82.8	81.5	77.9	88.9	95.1	94.6	98.4	97.4
georgian	70.6	68.8	17.2	58.9	92.1	91.6	82.9	92.5	93.9	93.7	98.5	98.4
greek	13.6	24.4	2.0	12.0	15.2	59.9	44.3	56.6	16.5	77.6	81.7	83.3
greenlandic	50.0	54.0	27.5	57.5	72.0	66.0	75.0	85.0				
haida	29.0	14.0	5.0	23.0	61.0	62.0	50.0	52.0	66.0	59.0	53.0	52.0
hebrew	24.4	25.5	4.1	13.8	38.1	50.1	76.3	76.3	53.7	61.7	98.1	97.2
hindi	31.8	28.6	23.9	65.6	86.5	85.2	94.3	95.1	93.0	92.0	98.6	97.5
hungarian	17.4	27.5	0.9	12.1	44.4	51.1	47.3	53.1	68.8	69.5	77.5	63.5
icelandic	35.6	38.9	6.5	14.9	58.9	62.4	52.3	61.3	76.9	75.1	84.3	78.7
ingrian	20.0	26.0	27.5	20.0	46.0	50.0	80.0	75.0				
irish	31.6	34.2	3.7	20.9	37.0	48.9	42.6	57.7	39.4	60.1	83.0	77.2
italian	40.5	42.2	3.3	41.3	72.5	84.8	81.3	91.1	77.5	94.4	97.9	95.4
kabardian	72.0	72.0	51.0	83.0	83.0	78.0	95.0	95.0	86.0	81.0	96.0	96.0
karelian	24.0	26.0	20.0	67.5	42.0	46.0	95.0	97.5				
kashubian	60.0	50.0	12.5	57.5	68.0	56.0	85.0	92.5				
kazakh	26.0	32.0	52.5	47.5	50.0	48.0	72.5	77.5				
khakas	26.0	30.0	27.5	65.0	84.0	84.0	85.0	92.5				
khaling	3.1	2.0	4.6	11.2	17.9	15.6	77.3	86.4	53.7	47.9	99.6	98.4

Continued on next page

Table D.1 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
kurmanji	82.7	86.5	0.0	58.4	85.2	88.4	83.7	88.2	92.9	91.9	92.8	91.4
ladin	58.0	52.0	30.0	52.0	86.0	83.0	88.0	95.0	92.0	92.0	98.0	98.0
latin	16.0	14.5	0.8	5.4	37.6	29.7	25.2	36.2	47.6	39.7	70.1	55.5
latvian	52.2	48.0	4.1	18.3	85.5	88.2	60.5	82.4	92.8	93.1	94.8	94.8
lithuanian	23.3	18.3	0.8	5.6	52.2	49.4	33.7	51.6	64.2	64.0	86.2	84.1
livonian	28.0	29.0	1.0	27.0	51.0	53.0	69.0	77.0	67.0	66.0	92.0	92.0
lower-sorbian	32.1	36.9	2.9	19.3	68.9	79.1	64.1	81.4	88.1	87.5	95.2	94.8
macedonian	49.8	45.2	5.1	37.7	82.6	85.3	75.7	89.8	91.2	92.1	96.4	95.3
maltese	9.0	16.0	0.0	23.0	20.0	25.0	87.0	93.0	16.0	26.0	97.0	98.0
mapudungun	64.0	66.0	57.5	95.0	82.0	86.0	97.5	97.5				
middle-french	76.9	75.7	10.1	67.2	90.3	90.9	89.2	93.0	95.1	93.6	98.8	96.3
middle-high-german	38.0	50.0	35.0	67.5	54.0	54.0	97.5	97.5				
murrinhpatha	6.0	10.0	25.0	35.0	20.0	20.0	95.0	90.0				
navajo	16.6	16.8	2.0	13.8	30.4	29.1	35.8	41.5	39.0	37.7	82.5	76.0
neapolitan	79.0	74.0	25.0	65.0	94.0	93.0	91.0	95.0	95.0	95.0	95.0	95.0
norman	30.0	28.0	45.0	60.0	46.0	32.0	77.5	80.0				
northern-sami	16.4	11.4	2.1	11.6	34.8	32.8	43.2	60.7	62.3	61.5	93.4	88.0
norwegian-bokmaal	67.8	72.2	13.8	54.8	80.7	82.4	78.0	76.5	91.0	90.0	88.9	77.0
norwegian-nynorsk	48.9	56.0	11.9	37.6	61.1	62.7	52.5	57.0	74.8	73.7	84.0	75.8
occitan	72.0	69.0	15.0	55.0	92.0	87.0	94.0	98.0	96.0	93.0	100.0	100.0
old-armenian	31.0	30.4	1.5	14.8	67.3	70.8	48.9	69.3	79.2	81.1	86.0	85.1

Continued on next page

Table D.1 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
old-church-slavonic	39.0	39.0	11.0	29.0	76.0	71.0	74.0	78.0	80.0	70.0	92.0	96.0
old-french	32.5	29.0	4.9	35.4	63.1	65.7	65.0	68.9	80.7	81.1	87.5	84.8
old-irish	8.0	8.0	5.0	5.0	16.0	14.0	20.0	32.5				
old-saxon	22.8	16.0	2.7	5.2	39.0	34.5	63.0	68.0	60.1	52.9	95.3	94.6
pashto	35.0	33.0	8.0	21.0	69.0	65.0	69.0	75.0	72.0	70.0	100.0	98.0
persian	4.3	28.0	2.8	35.7	4.8	71.9	82.1	85.7	6.1	84.3	96.0	95.4
portuguese	62.6	61.7	6.9	31.0	92.4	91.2	78.2	92.5	96.7	96.3	97.6	97.5
quechua	15.9	10.3	3.2	31.2	70.9	50.4	52.0	55.9	95.1	89.1	56.3	56.0
romanian	44.8	42.5	3.2	30.3	69.4	71.1	59.7	72.3	79.8	77.4	84.6	83.1
sanskrit	33.7	40.5	4.8	42.7	59.7	79.8	67.9	80.7	80.6	84.3	88.0	88.3
scottish-gaelic	46.0	46.0	25.0	50.0	50.0	52.0	80.0	90.0				
serbo-croatian	21.7	20.0	1.3	25.4	68.2	66.6	52.9	74.1	83.0	85.7	85.2	86.9
slovak	37.7	48.1	3.3	23.8	71.1	73.5	61.3	70.6	83.1	81.8	90.0	89.9
slovene	32.3	34.1	13.7	25.9	72.3	73.5	63.4	86.0	85.1	83.5	95.2	93.8
sorani	19.3	17.9	1.2	15.6	51.7	49.1	60.3	71.4	63.6	62.8	88.0	87.7
spanish	61.8	57.4	4.9	46.7	86.3	85.7	84.3	90.3	92.4	94.8	97.1	95.8
swahili	32.0	34.0	27.0	66.0	73.0	79.0	94.0	93.0	71.0	69.0	100.0	100.0
swedish	51.1	60.8	7.8	39.9	76.5	77.3	62.2	68.0	84.7	84.4	86.1	76.2
tatar	52.0	72.0	17.0	53.0	89.0	89.0	94.0	87.0	95.0	95.0	100.0	99.0
telugu	70.0	70.0	40.0	82.5								
tibetan	34.0	32.0	32.5	42.5	36.0	32.0	37.5	52.5				

Continued on next page

Table D.1 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
turkish	13.2	11.3	1.1	28.5	32.8	41.8	71.4	68.3	73.2	75.4	91.8	87.0
turkmen	34.0	68.0	37.5	60.0	68.0	74.0	87.5	92.5				
ukrainian	38.7	46.9	6.7	23.3	74.1	74.7	55.3	71.3	86.3	84.8	89.9	87.1
urdu	32.7	29.6	24.9	57.8	87.6	85.6	91.5	95.0	95.9	94.7	97.4	97.6
uzbek	52.0	30.0	47.0	74.0	96.0	93.0	78.0	78.0	96.0	93.0	78.0	78.0
venetian	71.8	71.3	16.6	42.3	89.1	87.3	91.6	93.1	93.0	91.6	99.6	99.0
votic	17.0	16.0	11.0	13.0	34.0	36.0	68.0	76.0	34.0	35.0	78.0	78.0
welsh	30.0	25.0	11.0	30.0	58.0	60.0	83.0	88.0	72.0	68.0	95.0	95.0
west-frisian	50.0	46.0	8.0	40.0	65.0	61.0	86.0	93.0	67.0	63.0	91.0	95.0
yiddish	78.0	79.0	6.0	60.0	87.0	88.0	83.0	92.0	94.0	86.0	98.0	99.0
zulu	15.6	14.9	11.0	33.3	52.8	62.8	81.6	86.7	68.4	77.0	99.2	97.7
dutch	50.8	53.5	7.8	24.1	72.4	74.0	73.5	79.4	87.7	86.8	96.2	95.1
english	77.6	83.2	28.5	56.4	90.5	91.4	85.7	88.0	95.9	95.4	95.6	93.6
french	59.0	56.6	3.9	37.7	73.2	72.1	71.9	71.6	83.0	83.0	83.7	73.5
german	49.2	50.9	10.7	11.5	71.7	74.2	66.0	71.1	81.1	81.6	88.4	82.0
kannada	33.0	36.0	9.0	27.0	55.0	63.0	83.0	90.0	66.0	66.0	95.0	95.0
middle-low-german	18.0	12.0	22.5	25.0	38.0	16.0	90.0	92.5				
north-frisian	24.0	27.0	11.0	27.0	25.0	26.0	85.0	82.0	24.0	37.0	94.0	95.0
old-english	17.6	11.4	4.3	12.7	27.8	22.2	38.3	53.3	40.9	33.1	83.8	79.5
polish	40.4	41.3	1.8	13.9	73.5	76.1	60.0	76.1	87.1	87.1	88.1	89.5
russian	43.4	46.1	1.8	11.5	76.4	79.7	54.4	76.5	86.5	87.6	89.2	87.7

Continued on next page

Table D.1 – continued from previous page

Language	**Accuracy**											
	low				**medium**				**high**			
	M	H	S	S-Aug	M	H	S	S-Aug	M	H	S	S-Aug
Average	38.3	39.6	13.1	36.9	61.8	64.1	71.3	78.5	74.7	77.2	90.9	89.1

Table D.2: SIGMORPHON 2018 Shared Task results on accuracy of different sizes of training data. Number of submissions: *low* (28), *medium* (21), *high* (22). The * mark means that the number is not comparable as there is no or only partial submissions of languages.

	Training data size		
	low	**medium**	**high**
Holistic+`dev` (ours)	58.3	67.2	77.2
UZH-02	57.2	86.4	96.0
UZH-01	57.2	86.6	96.0
UA-08	53.2	-	-
HYDERABAD-02	52.6	84.2	94.4
HYDERABAD-01	49.8	82.9	94.4
Hybrid	44.1	77.4	91.1
MSU-02	41.6	69.5	82.7
HAMBURG-01	40.3	74.0	77.5
Holistic (ours)	39.6	64.1	77.2
Baseline	38.9	63.5	77.4
MSU-04	31.4	76.4	91.9
MSU-03	25.9	75.7	90.5
VARANASI-01	23.3	70.2	91.7
AXSEMANTICS-02*	14.9	60.0	74.8
OSLO-02	4.4	29.3	56.6
BME-01	3.7	67.4	93.9
BME-03	3.6	67.4	94.0
KUCST-01*	2.8	32.3	54.4
BME-02	2.4	67.3	94.7
OSLO-03	1.4	31.0	63.1
OSLO-01	0.0	21.0	49.5

- UZH: University of Zurich, Switzerland
- UA: University of Alberta, Canada
- HYDERABAD: IIT Hyderabad, India
- MSU: University of Moscow, Russia
- HAMBURG: University of Hamburg
- VARANASI: IIT Varanasi, India
- AXSEMANTICS: AX Semantics, Germany
- OSLO: University of Oslo (Norway), University of Tuebingen (Germany)
- KUCST: University of Copenhagen, Denmark
- BME: Budapest University of Technology and Economics, Hungary

Table D.3: Accuracy at morphological analysis task in each language for morpheme-based system (**M**), holistic approach(**H**), our neural approaches: (**N-sml**) and (**N-fcs**).

Language	**Accuracy**											
	low				**medium**				**high**			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
adyghe	68.7	65.6	17.0	10.9	69.7	69.3	81.6	86.4	78.5	77.9	99.2	99.4
albanian	3.7	19.5	0.1	5.5	4.2	35.8	76.6	87.2	6.8	37.6	97.3	98.7
arabic	18.0	21.3	0.0	0.0	23.5	27.3	50.5	44.5	27.6	31.5	79.3	84.8
armenian	48.4	41.9	0.0	0.3	64.4	68.9	80.4	80.7	68.1	72.1	90.0	92.4
asturian	60.9	60.0	0.9	5.4	86.2	84.4	86.6	93.0	94.0	93.6	96.9	98.5
azeri	53.0	32.0	19.0	15.0	77.0	70.0	89.0	94.0	89.0	88.0	99.0	100.0
bashkir	69.4	63.1	26.5	0.0	93.5	93.1	95.1	94.7	96.7	96.7	98.5	98.7
basque	0.6	0.2	0.0	56.7	1.4	2.1	84.4	87.4	4.8	7.1	89.7	89.6
belarusian	11.3	36.4	0.2	0.9	36.8	58.5	61.6	66.3	71.3	75.8	93.1	96.8
bengali	41.0	46.0	17.0	33.0	71.0	69.0	95.0	97.0	82.0	83.0	91.0	100.0
breton	11.0	13.0	40.0	72.0	34.0	34.0	99.0	99.0	40.0	41.0	100.0	100.0
bulgarian	28.4	28.3	0.0	3.4	53.0	53.7	59.2	66.6	66.5	67.2	91.3	93.9
catalan	57.2	53.5	0.0	0.2	79.7	80.8	80.6	84.7	89.3	90.3	94.3	94.5
classical-syriac	85.0	89.0	39.0	41.0	90.0	91.0	89.0	96.0	91.0	95.0	99.0	99.0
cornish	8.0	14.0	0.0	82.0	20.0	18.0	70.0	86.0				
crimean-tatar	88.0	89.0	18.0	15.0	94.0	94.0	90.0	92.0	96.0	95.0	98.0	96.0
czech	47.8	47.8	0.0	0.0	70.8	73.0	70.5	75.6	78.2	78.4	83.9	87.7
danish	51.9	52.9	14.7	14.8	65.7	66.6	64.7	66.6	80.0	78.9	86.4	86.6
estonian	25.4	22.1	0.1	0.0	46.9	51.0	72.3	75.1	60.8	61.3	94.6	96.2

Continued on next page

Table D.3 – continued from previous page

Language	**Accuracy**											
	low				**medium**				**high**			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
faroese	30.9	32.0	0.0	5.9	44.7	47.6	34.0	46.7	66.1	65.6	69.1	78.6
finnish	3.9	14.7	0.1	0.0	31.9	36.1	50.6	64.2	46.7	50.6	78.1	79.8
friulian	75.0	67.0	14.0	36.0	87.0	88.0	86.0	99.0	96.0	94.0	97.0	100.0
galician	52.0	50.7	0.0	1.3	72.7	73.0	76.1	83.8	84.8	84.5	93.9	96.5
georgian	61.5	61.9	13.9	3.3	75.4	77.3	75.9	77.8	81.8	81.9	84.8	92.5
greek	10.9	26.1	0.0	7.0	11.3	46.0	50.0	56.4	14.3	57.1	66.8	77.2
greenlandic	42.0	32.0	84.0	90.0	48.0	52.0	100.0	98.0				
haida	50.0	14.0	0.0	97.0	80.0	63.0	98.0	100.0	81.0	74.0	100.0	100.0
hebrew	26.1	22.0	0.0	1.7	34.4	34.6	59.2	65.8	47.7	46.3	94.4	94.8
hindi	53.5	52.2	6.4	4.9	81.2	91.4	97.2	96.8	88.6	96.3	99.9	100.0
hungarian	53.9	41.5	0.5	1.2	81.9	78.9	80.7	85.8	92.3	92.2	93.1	92.8
icelandic	32.2	32.3	0.0	10.8	46.6	46.7	44.2	43.8	67.0	67.5	69.0	75.3
ingrian	16.0	14.0	36.0	84.0	42.0	36.0	100.0	100.0				
irish	28.6	23.3	0.7	0.5	37.3	33.3	53.8	61.6	40.0	37.3	77.9	85.3
italian	40.8	43.2	0.0	0.4	53.8	60.9	75.5	80.9	59.2	68.2	91.5	91.1
kabardian	64.0	63.0	39.0	32.0	72.0	69.0	98.0	98.0	80.0	79.0	98.0	99.0
karelian	46.0	42.0	92.0	100.0	74.0	70.0	98.0	100.0				
kashubian	64.0	64.0	80.0	76.0	84.0	86.0	100.0	98.0				
kazakh	68.0	62.0	94.0	98.0	82.0	68.0	100.0	100.0				
khakas	62.0	58.0	60.0	72.0	90.0	96.0	100.0	98.0				
khaling	4.5	7.6	0.0	8.2	11.1	25.7	63.0	67.9	16.5	48.5	87.6	89.2

Continued on next page

Table D.3 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
kurmanji	78.3	78.7	0.0	10.0	83.4	84.9	69.6	81.7	84.6	88.0	88.4	90.6
ladin	60.0	60.0	0.0	40.0	81.0	81.0	92.0	95.0	93.0	93.0	99.0	99.0
latin	17.6	15.3	0.2	0.2	38.4	32.2	62.4	68.4	51.5	45.9	76.6	81.0
latvian	49.6	50.1	0.0	14.7	70.7	70.4	58.4	58.2	77.9	78.4	81.8	83.3
lithuanian	19.4	19.4	0.0	0.9	48.2	48.6	60.5	65.1	64.0	64.6	89.7	91.9
livonian	37.0	31.0	9.0	5.0	60.0	60.0	92.0	93.0	74.0	74.0	96.0	94.0
lower-sorbian	45.4	43.2	0.6	2.7	56.6	59.0	52.5	68.7	76.5	77.4	95.2	95.1
macedonian	48.5	46.7	0.1	14.7	66.5	68.6	64.3	69.9	81.3	81.1	81.3	83.2
maltese	16.0	14.0	8.0	19.0	26.0	25.0	93.0	96.0	29.0	35.0	92.0	96.0
mapudungun	40.0	40.0	90.0	100.0	60.0	64.0	100.0	100.0				
middle-french	70.3	66.5	0.0	9.8	78.9	78.8	82.0	90.4	88.1	88.1	97.5	98.5
middle-high-german	58.0	58.0	50.0	90.0	84.0	82.0	98.0	100.0				
murrinhpatha	18.0	20.0	4.0	48.0	28.0	38.0	60.0	70.0				
navajo	13.0	13.5	0.0	0.9	24.2	22.0	22.7	36.9	31.2	29.8	83.4	82.9
neapolitan	65.0	75.0	49.0	91.0	88.0	87.0	99.0	100.0	87.0	85.0	100.0	100.0
norman	48.0	44.0	96.0	100.0	56.0	58.0	98.0	100.0				
northern-sami	14.8	11.5	0.0	0.7	29.5	29.0	43.9	57.3	55.6	56.1	79.5	82.7
norwegian-bokmaal	64.6	61.9	6.9	19.4	70.6	71.4	62.9	66.9	80.1	80.1	81.7	77.0
norwegian-nynorsk	61.5	59.5	13.4	17.5	65.8	65.5	53.8	54.4	73.4	73.6	75.9	72.7
occitan	66.0	63.0	3.0	40.0	82.0	83.0	95.0	95.0	92.0	91.0	98.0	98.0
old-armenian	39.3	43.9	0.0	8.3	62.8	66.3	59.1	60.3	73.7	75.6	80.6	84.7

Continued on next page

Table D.3 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
old-church-slavonic	45.0	43.0	20.0	42.0	69.0	69.0	77.0	74.0	73.0	71.0	78.0	77.0
old-french	28.4	26.3	0.0	3.6	47.8	48.1	39.6	53.7	61.4	62.0	67.1	79.1
old-irish	8.0	12.0	4.0	28.0	24.0	26.0	90.0	92.0				
old-saxon	24.4	20.8	0.0	0.9	44.6	43.8	64.4	68.1	81.7	80.1	92.8	95.1
pashto	29.0	33.0	4.0	4.0	55.0	57.0	66.0	83.0	65.0	64.0	92.0	97.0
persian	2.6	34.3	0.5	0.4	4.6	60.7	69.1	95.2	17.6	71.0	97.7	98.8
portuguese	66.7	66.4	0.1	24.3	84.4	85.8	83.1	89.3	91.7	92.0	95.7	95.6
quechua	27.6	9.8	6.0	9.6	72.5	53.6	86.6	89.1	82.7	88.2	94.0	95.5
romanian	39.8	41.9	0.0	0.1	54.1	58.2	52.9	62.2	61.3	67.7	79.7	83.6
sanskrit	66.8	66.2	0.0	0.7	75.4	76.8	80.0	82.8	86.7	87.3	97.2	97.6
scottish-gaelic	70.0	68.0	50.0	42.0	90.0	92.0	98.0	98.0				
serbo-croatian	32.7	33.6	0.0	0.7	55.9	62.2	70.7	73.2	64.1	68.7	77.3	79.1
slovak	49.9	53.1	9.8	7.3	63.7	63.6	67.8	66.3	83.3	83.8	89.8	96.5
slovene	45.8	45.7	2.9	7.6	69.4	69.7	73.4	76.1	85.8	84.9	93.2	95.7
sorani	11.5	12.6	0.0	11.2	21.2	31.3	82.3	87.4	26.2	27.2	98.9	98.6
spanish	51.3	52.8	0.0	1.4	72.2	73.4	78.3	87.4	78.7	80.1	92.7	94.5
swahili	40.0	40.0	0.0	51.0	80.0	86.0	91.0	95.0	94.0	91.0	100.0	100.0
swedish	53.6	58.3	1.9	26.2	71.1	73.2	63.0	63.6	80.9	80.8	77.0	79.0
tatar	91.0	90.0	9.0	19.0	96.0	95.0	82.0	90.0	98.0	98.0	99.0	99.0
telugu	80.0	80.0	80.0	100.0								
tibetan	48.0	56.0	60.0	76.0	52.0	58.0	66.0	86.0				

Continued on next page

Table D.3 – continued from previous page

Language	Accuracy											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
turkish	31.2	17.6	0.0	0.6	67.1	54.6	75.7	82.4	83.2	82.8	91.2	92.7
turkmen	88.0	86.0	56.0	84.0	96.0	96.0	98.0	98.0				
ukrainian	41.7	42.8	1.8	6.2	58.9	60.3	58.5	53.0	80.1	79.6	93.7	95.1
urdu	46.6	44.9	24.9	54.8	79.1	89.5	92.2	95.6	90.3	92.9	99.7	100.0
uzbek	73.0	35.0	88.0	99.0	94.0	98.0	100.0	100.0	94.0	98.0	100.0	100.0
venetian	75.0	79.6	0.8	30.6	86.7	87.7	87.6	93.1	95.1	95.3	94.2	98.4
votic	27.0	15.0	70.0	75.0	35.0	34.0	99.0	97.0	42.0	40.0	98.0	98.0
welsh	18.0	16.0	3.0	5.0	34.0	32.0	84.0	94.0	52.0	51.0	98.0	99.0
west-frisian	49.0	49.0	0.0	6.0	87.0	87.0	88.0	98.0	89.0	90.0	71.0	96.0
yiddish	59.0	66.0	13.0	4.0	73.0	71.0	58.0	77.0	94.0	93.0	96.0	97.0
zulu	28.0	25.7	0.0	0.7	68.0	68.6	68.6	73.7	81.0	82.1	94.6	96.9
dutch	54.6	54.7	0.3	14.7	66.8	67.1	26.2	61.1	83.4	82.6	78.2	88.3
english	74.4	72.5	5.7	49.8	87.4	87.3	56.9	84.7	94.5	94.3	77.0	93.9
french	55.7	53.7	0.2	0.0	70.4	71.7	67.1	75.7	80.8	80.7	84.4	85.9
german	53.5	52.1	0.3	7.9	63.7	64.6	65.0	63.9	78.6	80.4	83.8	81.0
kannada	46.0	43.0	0.0	12.0	67.0	67.0	62.0	70.0	71.0	71.0	73.0	75.0
middle-low-german	24.0	24.0	0.0	50.0	84.0	84.0	94.0	98.0				
north-frisian	35.0	48.0	18.0	56.0	49.0	71.0	92.0	93.0	50.0	78.0	96.0	94.0
old-english	22.6	18.9	0.0	10.6	37.3	36.6	51.0	62.1	68.6	68.6	80.6	87.1
polish	43.7	39.5	0.0	4.0	60.7	61.5	58.2	64.1	72.9	73.5	74.0	78.6
russian	42.3	43.5	0.5	7.9	62.7	65.3	60.8	67.4	74.4	76.7	79.2	80.6

Continued on next page

Table D.3 – continued from previous page

Language	**Accuracy**											
	low				**medium**				**high**			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
Average	43.6	42.8	15.3	26.7	60.9	63.1	75.3	80.8	71.0	73.7	89.1	91.6

Table D.4: Average Levenshtein distance at morphological analysis task in each language for morpheme-based system (**M**), holistic approach(**H**), our neural approaches: (**N-`sml`**) and (**N-`fcs`**).

Language	**Average Levenshtein distance**											
	low				**medium**				**high**			
	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`
adyghe	0.32	0.39	2.41	2.86	0.32	0.34	0.22	0.15	0.22	0.23	0.01	0.01
albanian	3.36	5.29	5.34	4.17	2.67	4.75	0.51	0.23	2.11	4.45	0.05	0.02
arabic	2.85	3.14	6.43	6.16	2.49	2.83	1.98	2.07	2.20	2.60	0.81	0.56
armenian	1.37	1.75	5.81	4.59	0.98	0.94	0.30	0.34	0.85	0.86	0.16	0.12
asturian	0.69	0.84	3.81	3.00	0.25	0.32	0.24	0.13	0.11	0.12	0.05	0.03
azeri	1.37	2.48	2.77	3.41	0.36	0.81	0.24	0.18	0.15	0.21	0.02	0.00
bashkir	0.37	1.12	1.70	4.13	0.12	0.15	0.13	0.31	0.07	0.08	0.03	0.03
basque	4.57	6.18	3.06	1.41	3.48	5.50	0.42	0.36	2.10	4.67	0.28	0.27
belarusian	2.09	1.63	5.89	4.68	1.45	1.05	0.65	0.58	0.83	0.64	0.12	0.06
bengali	1.18	1.39	2.58	2.10	0.60	0.64	0.08	0.05	0.34	0.31	0.13	0.00
breton	2.48	2.91	1.73	0.67	1.86	1.91	0.05	0.01	1.59	1.67	0.00	0.00
bulgarian	1.52	1.84	6.40	4.12	1.14	1.16	0.68	0.54	0.84	0.91	0.14	0.10
catalan	0.89	1.11	4.97	4.56	0.49	0.49	0.39	0.34	0.29	0.27	0.15	0.13
classical-syriac	0.19	0.14	1.09	1.29	0.11	0.10	0.11	0.04	0.10	0.06	0.01	0.01
cornish	2.52	3.02	3.34	0.18	1.58	2.14	0.38	0.14				
crimean-tatar	0.23	0.27	2.00	2.11	0.10	0.14	0.18	0.15	0.09	0.12	0.05	0.09
czech	1.59	1.95	6.00	5.74	0.97	0.75	0.48	0.44	0.85	0.75	0.25	0.23
danish	0.65	0.68	2.90	2.27	0.48	0.48	0.57	0.61	0.30	0.32	0.22	0.21
estonian	2.43	2.84	5.90	5.40	1.96	1.92	0.62	0.53	1.58	1.80	0.10	0.08

Continued on next page

Table D.4 – continued from previous page

Language	Average Levenshtein distance											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
faroese	1.33	1.31	6.10	4.12	1.05	1.00	1.26	0.98	0.66	0.67	0.53	0.40
finnish	3.88	4.30	8.66	7.77	3.00	3.30	1.38	0.72	2.61	2.88	0.59	0.43
friulian	0.37	0.80	2.39	1.45	0.16	0.21	0.17	0.01	0.06	0.13	0.03	0.00
galician	0.83	1.01	4.50	3.52	0.51	0.56	0.40	0.26	0.28	0.30	0.12	0.08
georgian	0.55	0.63	2.30	3.34	0.37	0.39	0.46	0.38	0.27	0.28	0.20	0.11
greek	2.63	3.05	6.65	3.45	2.42	2.26	1.07	0.94	2.12	1.78	0.61	0.42
greenlandic	0.78	1.32	0.46	0.28	0.68	0.68	0.00	0.04				
haida	1.67	7.02	5.31	0.06	0.35	2.35	0.07	0.00	0.29	0.91	0.00	0.00
hebrew	1.16	1.43	3.33	3.31	0.94	1.06	0.62	0.53	0.66	0.78	0.09	0.08
hindi	1.21	2.16	3.30	3.40	0.45	0.29	0.12	0.05	0.33	0.15	0.00	0.00
hungarian	0.71	1.60	6.55	5.49	0.33	0.50	0.53	0.53	0.18	0.18	0.26	0.17
icelandic	1.10	1.24	5.77	2.81	0.87	0.87	0.92	1.11	0.53	0.52	0.46	0.40
ingrian	1.62	2.14	1.62	0.26	0.78	1.10	0.00	0.00				
irish	2.10	3.54	6.18	5.67	1.85	2.75	1.06	0.86	1.69	2.56	0.45	0.29
italian	1.92	1.94	7.12	5.44	1.56	1.41	0.39	0.33	1.39	1.20	0.14	0.15
kabardian	0.37	0.40	1.30	2.28	0.28	0.32	0.02	0.02	0.20	0.27	0.02	0.01
karelian	1.32	1.40	0.26	0.00	0.44	0.60	0.06	0.00				
kashubian	0.42	0.42	0.66	1.14	0.18	0.16	0.00	0.02				
kazakh	0.40	1.12	0.14	0.04	0.24	0.82	0.00	0.00				
khakas	0.40	1.18	0.98	0.64	0.12	0.08	0.00	0.08				
khaling	2.96	3.67	3.27	1.94	2.28	2.48	0.48	0.39	1.81	1.44	0.15	0.13

Continued on next page

Table D.4 – continued from previous page

Language	Average Levenshtein distance											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
kurmanji	0.54	0.57	5.65	3.81	0.41	0.39	0.84	0.65	0.36	0.29	0.36	0.39
ladin	0.73	0.86	4.23	1.78	0.32	0.38	0.12	0.08	0.12	0.12	0.01	0.01
latin	2.19	3.00	6.59	6.21	1.09	1.90	0.72	0.60	0.74	1.18	0.39	0.33
latvian	0.95	0.99	6.53	2.62	0.61	0.62	0.92	1.01	0.46	0.46	0.32	0.29
lithuanian	1.85	2.23	7.48	4.99	0.94	1.05	0.84	0.76	0.59	0.65	0.20	0.15
livonian	1.65	1.94	4.06	4.38	1.16	1.11	0.20	0.11	0.76	0.92	0.05	0.11
lower-sorbian	0.78	0.88	4.27	4.13	0.62	0.61	0.69	0.43	0.33	0.32	0.07	0.07
macedonian	0.69	0.96	5.15	1.92	0.42	0.43	0.50	0.52	0.26	0.26	0.25	0.23
maltese	1.87	2.06	3.44	3.08	1.21	1.57	0.13	0.07	1.07	1.17	0.12	0.06
mapudungun	0.72	0.96	0.18	0.00	0.40	0.42	0.00	0.00				
middle-french	0.62	0.73	4.42	2.63	0.43	0.47	0.35	0.18	0.24	0.24	0.03	0.02
middle-high-german	0.74	0.86	1.48	0.24	0.18	0.36	0.08	0.00				
murrinhpatha	1.46	2.60	1.90	0.90	1.14	1.64	0.80	0.64				
navajo	3.05	3.51	5.40	6.86	2.22	2.84	2.23	1.78	1.80	2.33	0.36	0.36
neapolitan	0.60	0.86	1.18	0.36	0.23	0.33	0.03	0.00	0.28	0.47	0.00	0.00
norman	1.48	1.72	0.04	0.00	1.06	1.52	0.12	0.00				
northern-sami	1.96	2.90	5.36	4.16	1.35	1.77	1.14	0.79	0.77	0.88	0.31	0.30
norwegian-bokmaal	0.46	0.55	3.00	2.21	0.40	0.40	0.56	0.56	0.27	0.28	0.25	0.34
norwegian-nynorsk	0.51	0.56	2.69	1.93	0.46	0.46	0.63	0.76	0.33	0.33	0.30	0.34
occitan	0.76	0.85	3.09	1.60	0.36	0.39	0.09	0.08	0.15	0.23	0.07	0.06
old-armenian	1.25	1.31	4.59	2.78	0.74	0.63	0.70	0.73	0.55	0.49	0.35	0.29

Continued on next page

Table D.4 – continued from previous page

Language	Average Levenshtein distance											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
old-church-slavonic	0.77	0.99	2.76	1.75	0.37	0.40	0.23	0.26	0.32	0.37	0.22	0.23
old-french	1.46	1.59	5.25	3.13	1.05	1.05	1.06	0.80	0.75	0.72	0.55	0.35
old-irish	3.98	3.88	4.20	3.34	3.00	2.94	0.42	0.36				
old-saxon	1.64	2.00	4.84	4.16	1.05	1.20	0.69	0.60	0.35	0.41	0.12	0.09
pashto	1.69	1.81	3.00	2.89	0.84	0.93	0.47	0.29	0.61	0.83	0.12	0.04
persian	3.81	2.56	4.55	5.19	3.04	1.61	0.70	0.09	1.45	1.19	0.04	0.02
portuguese	0.62	0.74	4.92	1.56	0.29	0.28	0.26	0.16	0.15	0.15	0.07	0.07
quechua	2.20	5.50	3.04	2.70	0.45	1.92	0.26	0.21	0.23	0.25	0.09	0.05
romanian	1.66	1.70	5.07	4.76	1.37	1.14	0.84	0.76	1.17	0.89	0.41	0.27
sanskrit	0.47	0.62	5.70	4.05	0.29	0.30	0.28	0.23	0.17	0.17	0.03	0.03
scottish-gaelic	0.52	0.68	2.38	3.72	0.18	0.16	0.02	0.02				
serbo-croatian	1.98	2.02	6.30	4.39	1.43	1.28	0.83	0.67	1.28	1.13	0.44	0.41
slovak	0.65	0.63	2.39	2.58	0.47	0.47	0.46	0.60	0.22	0.21	0.14	0.05
slovene	0.80	0.80	4.05	3.45	0.50	0.52	0.43	0.38	0.24	0.27	0.11	0.08
sorani	2.90	3.74	3.96	3.05	2.32	2.68	0.40	0.29	1.90	3.10	0.02	0.03
spanish	1.20	1.35	4.95	3.87	0.80	0.81	0.37	0.21	0.63	0.61	0.10	0.08
swahili	1.23	2.17	3.29	1.17	0.29	0.25	0.21	0.16	0.09	0.23	0.00	0.00
swedish	0.74	0.75	4.41	2.14	0.47	0.43	0.64	0.97	0.30	0.30	0.40	0.35
tatar	0.17	0.22	2.50	1.74	0.08	0.11	0.32	0.21	0.04	0.04	0.02	0.01
telugu	0.42	0.62	0.52	0.00								
tibetan	0.72	0.62	0.76	0.54	0.62	0.58	0.60	0.30				

Continued on next page

Table D.4 – continued from previous page

Language	Average Levenshtein distance											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
turkish	2.49	4.08	5.44	5.79	0.81	1.78	0.56	0.55	0.30	0.33	0.14	0.15
turkmen	0.14	0.18	0.86	0.54	0.06	0.06	0.02	0.04				
ukrainian	0.94	0.92	6.14	3.15	0.69	0.65	0.68	0.73	0.36	0.35	0.09	0.08
urdu	1.29	2.32	2.15	1.20	0.55	0.32	0.19	0.12	0.37	0.37	0.00	0.00
uzbek	0.62	2.02	0.21	0.02	0.06	0.04	0.00	0.00	0.06	0.04	0.00	0.00
venetian	0.44	0.45	3.35	1.48	0.22	0.21	0.19	0.11	0.07	0.07	0.08	0.02
votic	1.45	2.05	0.65	0.61	0.99	1.27	0.02	0.07	0.92	1.09	0.04	0.04
welsh	1.53	1.76	3.82	4.35	1.12	1.26	0.33	0.22	0.80	0.83	0.06	0.07
west-frisian	1.05	1.19	4.34	2.88	0.23	0.20	0.30	0.04	0.20	0.16	0.56	0.06
yiddish	0.90	0.70	3.79	3.44	0.56	0.60	0.74	1.02	0.11	0.15	0.09	0.08
zulu	1.44	2.01	4.53	4.08	0.60	0.69	0.60	0.51	0.38	0.38	0.09	0.06
dutch	0.91	0.90	4.78	2.71	0.69	0.69	1.77	0.75	0.36	0.38	0.46	0.24
english	0.28	0.32	2.81	0.75	0.15	0.14	0.69	0.20	0.06	0.07	0.30	0.08
french	0.84	1.18	4.52	5.13	0.56	0.57	0.55	0.44	0.37	0.37	0.25	0.23
german	0.86	0.92	6.91	3.24	0.74	0.76	1.02	0.79	0.50	0.49	0.28	0.34
kannada	1.92	2.04	3.85	3.19	2.66	1.28	1.12	0.99	3.49	1.16	1.24	0.80
middle-low-german	1.86	1.66	4.20	1.58	0.32	0.44	0.16	0.04				
north-frisian	2.19	2.33	2.62	1.16	1.17	1.23	0.20	0.15	1.10	1.23	0.11	0.16
old-english	1.54	1.92	5.24	2.72	1.17	1.29	0.88	0.67	0.59	0.62	0.31	0.23
polish	1.42	1.52	6.56	3.64	0.96	0.91	0.82	0.71	0.77	0.69	0.47	0.41
russian	1.24	1.24	5.31	3.14	0.88	0.79	0.86	0.98	0.67	0.54	0.45	0.43

Continued on next page

Table D.4 – continued from previous page

Language	**Average Levenshtein distance**											
	low				**medium**				**high**			
	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`
Average	1.34	1.73	3.81	2.82	0.88	1.01	0.49	0.40	0.68	0.76	0.20	0.16

Table D.5: Precision at morphological analysis task in each language for morpheme-based system (**M**), holistic approach(**H**), our neural approaches: (**N-`sml`**) and (**N-`fcs`**).

Language	**Precision**											
	low				**medium**				**high**			
	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`
adyghe	0.89	0.81	0.94	0.97	0.96	0.92	0.97	0.98	0.96	0.92	0.99	0.99
albanian	0.57	0.50	0.49	0.47	0.66	0.62	0.89	0.86	0.74	0.68	0.94	0.94
arabic	0.58	0.51	0.28	0.47	0.66	0.58	0.76	0.80	0.69	0.62	0.88	0.89
armenian	0.70	0.67	0.48	0.56	0.78	0.72	0.88	0.87	0.84	0.79	0.93	0.93
asturian	0.78	0.74	0.58	0.70	0.87	0.83	0.88	0.89	0.87	0.86	0.90	0.90
azeri	0.73	0.66	0.73	0.80	0.88	0.83	0.95	0.94	0.92	0.90	0.97	0.96
bashkir	0.88	0.83	0.88	0.91	0.92	0.89	0.94	0.94	0.93	0.90	0.95	0.95
basque	0.63	0.59	0.51	0.48	0.76	0.69	0.95	0.96	0.83	0.79	0.98	0.98
belarusian	0.64	0.58	0.42	0.38	0.73	0.68	0.75	0.72	0.75	0.72	0.82	0.83
bengali	0.82	0.77	0.65	0.65	0.90	0.85	0.92	0.91	0.89	0.88	0.90	0.92
breton	0.90	0.86	0.80	0.84	0.93	0.91	0.96	0.95	0.93	0.93	0.95	0.96
bulgarian	0.75	0.68	0.38	0.50	0.80	0.75	0.86	0.86	0.87	0.84	0.92	0.91
catalan	0.84	0.78	0.55	0.68	0.91	0.88	0.92	0.92	0.92	0.90	0.93	0.93
classical-syriac	0.86	0.77	0.67	0.78	0.83	0.78	0.87	0.88	0.83	0.79	0.88	0.89
cornish	0.75	0.66	0.59	0.61	0.82	0.75	0.90	0.86				
crimean-tatar	0.86	0.82	0.83	0.90	0.93	0.85	0.93	0.92	0.93	0.90	0.97	0.94
czech	0.58	0.53	0.39	0.48	0.70	0.63	0.71	0.69	0.73	0.67	0.76	0.77
danish	0.84	0.76	0.85	0.76	0.88	0.85	0.91	0.91	0.92	0.89	0.94	0.95
estonian	0.75	0.70	0.57	0.63	0.83	0.80	0.90	0.90	0.85	0.82	0.95	0.95

Continued on next page

Table D.5 – continued from previous page

Language	Precision											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
faroese	0.60	0.57	0.48	0.49	0.69	0.63	0.69	0.70	0.76	0.71	0.80	0.79
finnish	0.48	0.57	0.54	0.49	0.76	0.71	0.82	0.86	0.82	0.77	0.93	0.93
friulian	0.81	0.78	0.64	0.49	0.91	0.87	0.89	0.89	0.85	0.85	0.91	0.90
galician	0.72	0.65	0.46	0.65	0.84	0.80	0.84	0.84	0.85	0.83	0.88	0.88
georgian	0.85	0.80	0.80	0.69	0.92	0.87	0.93	0.93	0.94	0.90	0.95	0.96
greek	0.42	0.53	0.43	0.42	0.45	0.64	0.71	0.75	0.53	0.69	0.82	0.83
greenlandic	0.88	0.91	0.85	0.91	0.91	0.91	0.91	0.93				
haida	0.81	0.77	0.55	0.75	0.90	0.88	0.97	0.99	0.95	0.95	0.98	0.99
hebrew	0.61	0.54	0.39	0.48	0.71	0.62	0.80	0.84	0.72	0.65	0.91	0.89
hindi	0.79	0.72	0.75	0.78	0.86	0.81	0.90	0.89	0.87	0.83	0.90	0.91
hungarian	0.75	0.71	0.65	0.60	0.87	0.80	0.89	0.87	0.93	0.89	0.95	0.94
icelandic	0.69	0.63	0.53	0.55	0.74	0.69	0.75	0.74	0.78	0.74	0.81	0.80
ingrian	0.90	0.83	0.88	0.85	0.95	0.88	0.95	0.96				
irish	0.62	0.50	0.60	0.51	0.63	0.59	0.80	0.79	0.68	0.63	0.85	0.86
italian	0.81	0.76	0.50	0.62	0.90	0.87	0.91	0.89	0.91	0.89	0.93	0.92
kabardian	0.94	0.86	0.96	0.95	0.93	0.91	0.99	0.99	0.96	0.93	0.99	0.98
karelian	0.83	0.79	0.55	0.89	0.87	0.86	0.91	0.91				
kashubian	0.79	0.78	0.72	0.81	0.79	0.79	0.78	0.84				
kazakh	0.84	0.85	0.89	0.95	0.90	0.90	0.94	0.96				
khakas	0.83	0.85	0.91	0.92	0.96	0.97	0.99	0.99				
khaling	0.64	0.60	0.54	0.70	0.74	0.68	0.80	0.83	0.74	0.69	0.86	0.86

Continued on next page

Table D.5 – continued from previous page

Language	Precision											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
kurmanji	0.84	0.77	0.58	0.68	0.91	0.85	0.92	0.93	0.93	0.88	0.96	0.96
ladin	0.76	0.72	0.51	0.67	0.82	0.77	0.85	0.83	0.78	0.78	0.84	0.84
latin	0.55	0.50	0.49	0.37	0.73	0.70	0.67	0.71	0.79	0.74	0.80	0.79
latvian	0.54	0.48	0.39	0.41	0.67	0.63	0.68	0.66	0.74	0.69	0.77	0.80
lithuanian	0.58	0.52	0.31	0.31	0.79	0.72	0.78	0.78	0.84	0.81	0.91	0.89
livonian	0.71	0.62	0.54	0.52	0.83	0.76	0.90	0.89	0.84	0.83	0.92	0.93
lower-sorbian	0.64	0.59	0.39	0.36	0.71	0.67	0.70	0.74	0.70	0.67	0.78	0.76
macedonian	0.75	0.70	0.51	0.38	0.86	0.80	0.85	0.86	0.90	0.86	0.90	0.91
maltese	0.79	0.73	0.78	0.85	0.80	0.73	0.93	0.92	0.81	0.73	0.93	0.92
mapudungun	0.92	0.88	0.86	0.89	0.93	0.93	0.97	0.98				
middle-french	0.85	0.77	0.51	0.66	0.89	0.86	0.91	0.92	0.89	0.88	0.92	0.92
middle-high-german	0.62	0.57	0.55	0.67	0.70	0.65	0.78	0.78				
murrinhpatha	0.69	0.60	0.56	0.55	0.71	0.66	0.85	0.84				
navajo	0.62	0.57	0.55	0.61	0.81	0.75	0.83	0.84	0.85	0.81	0.94	0.94
neapolitan	0.83	0.78	0.74	0.78	0.89	0.87	0.91	0.91	0.88	0.88	0.94	0.93
norman	0.72	0.70	0.68	0.77	0.77	0.77	0.49	0.77				
northern-sami	0.70	0.69	0.39	0.58	0.81	0.79	0.84	0.84	0.86	0.84	0.87	0.88
norwegian-bokmaal	0.79	0.76	0.68	0.71	0.86	0.83	0.85	0.83	0.88	0.86	0.90	0.90
norwegian-nynorsk	0.80	0.75	0.73	0.57	0.82	0.79	0.80	0.81	0.90	0.88	0.91	0.91
occitan	0.84	0.80	0.73	0.84	0.93	0.92	0.93	0.94	0.92	0.92	0.95	0.95
old-armenian	0.57	0.52	0.40	0.48	0.72	0.66	0.73	0.75	0.76	0.70	0.78	0.80

Continued on next page

Table D.5 – continued from previous page

Language	Precision											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
old-church-slavonic	0.69	0.64	0.58	0.57	0.70	0.67	0.72	0.69	0.67	0.63	0.69	0.70
old-french	0.80	0.75	0.49	0.67	0.88	0.82	0.89	0.90	0.90	0.86	0.92	0.92
old-irish	0.59	0.52	0.53	0.56	0.71	0.60	0.74	0.72				
old-saxon	0.56	0.55	0.36	0.39	0.69	0.64	0.72	0.72	0.77	0.74	0.82	0.83
pashto	0.62	0.55	0.53	0.51	0.77	0.67	0.84	0.82	0.77	0.70	0.85	0.83
persian	0.54	0.71	0.63	0.71	0.58	0.80	0.93	0.91	0.70	0.84	0.94	0.94
portuguese	0.81	0.73	0.62	0.64	0.87	0.83	0.86	0.88	0.88	0.85	0.88	0.89
quechua	0.68	0.60	0.56	0.52	0.86	0.83	0.90	0.90	0.90	0.85	0.92	0.93
romanian	0.59	0.58	0.50	0.46	0.71	0.65	0.79	0.80	0.76	0.70	0.86	0.84
sanskrit	0.60	0.50	0.35	0.37	0.63	0.55	0.72	0.68	0.68	0.63	0.80	0.79
scottish-gaelic	0.55	0.44	0.52	0.52	0.56	0.56	0.56	0.68				
serbo-croatian	0.53	0.48	0.38	0.40	0.67	0.60	0.70	0.71	0.69	0.63	0.75	0.76
slovak	0.69	0.59	0.58	0.49	0.75	0.69	0.74	0.74	0.73	0.69	0.80	0.80
slovene	0.55	0.51	0.44	0.34	0.64	0.58	0.64	0.62	0.65	0.62	0.69	0.68
sorani	0.68	0.64	0.51	0.65	0.80	0.75	0.91	0.91	0.83	0.81	0.99	0.98
spanish	0.82	0.73	0.67	0.80	0.89	0.87	0.92	0.92	0.92	0.89	0.94	0.94
swahili	0.74	0.68	0.60	0.75	0.86	0.83	0.88	0.92	0.87	0.86	0.92	0.91
swedish	0.69	0.65	0.61	0.48	0.76	0.71	0.78	0.76	0.82	0.77	0.83	0.84
tatar	0.90	0.87	0.77	0.82	0.97	0.90	0.94	0.97	0.96	0.91	0.98	0.96
telugu	0.89	0.85	0.86	0.89								
tibetan	0.56	0.50	0.62	0.63	0.53	0.52	0.56	0.57				

Continued on next page

Table D.5 – continued from previous page

Language	Precision											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
turkish	0.61	0.56	0.49	0.60	0.82	0.78	0.89	0.89	0.89	0.86	0.93	0.92
turkmen	0.85	0.85	0.89	0.95	0.96	0.96	0.98	0.99				
ukrainian	0.65	0.59	0.54	0.51	0.77	0.70	0.77	0.77	0.77	0.73	0.81	0.82
urdu	0.72	0.65	0.68	0.75	0.81	0.75	0.84	0.85	0.80	0.80	0.88	0.88
uzbek	0.84	0.80	0.78	0.95	0.86	0.88	0.94	0.95	0.87	0.88	0.95	0.95
venetian	0.77	0.71	0.50	0.46	0.79	0.75	0.81	0.82	0.77	0.76	0.83	0.83
votic	0.87	0.83	0.83	0.90	0.94	0.91	0.97	0.98	0.94	0.91	0.97	0.97
welsh	0.84	0.82	0.73	0.84	0.91	0.89	0.92	0.95	0.89	0.89	0.92	0.95
west-frisian	0.66	0.62	0.51	0.63	0.67	0.66	0.71	0.76	0.68	0.67	0.64	0.72
yiddish	0.55	0.51	0.54	0.46	0.59	0.57	0.61	0.63	0.55	0.56	0.69	0.72
zulu	0.56	0.52	0.46	0.40	0.77	0.73	0.80	0.81	0.80	0.77	0.90	0.89
dutch	0.58	0.51	0.45	0.53	0.64	0.58	0.58	0.64	0.64	0.60	0.71	0.71
english	0.91	0.89	0.92	0.89	0.94	0.92	0.92	0.98	0.95	0.93	0.92	0.92
french	0.79	0.72	0.65	0.66	0.87	0.83	0.89	0.90	0.89	0.86	0.90	0.91
german	0.60	0.51	0.55	0.52	0.61	0.54	0.70	0.72	0.67	0.61	0.75	0.76
kannada	0.73	0.69	0.43	0.57	0.84	0.81	0.89	0.90	0.89	0.84	0.94	0.92
middle-low-german	0.50	0.51	0.46	0.49	0.61	0.59	0.74	0.77				
north-frisian	0.69	0.65	0.68	0.70	0.71	0.71	0.77	0.80	0.70	0.67	0.77	0.78
old-english	0.55	0.53	0.28	0.39	0.60	0.55	0.61	0.63	0.67	0.64	0.71	0.72
polish	0.60	0.55	0.41	0.36	0.73	0.67	0.74	0.71	0.78	0.72	0.81	0.81
russian	0.67	0.60	0.42	0.35	0.77	0.70	0.77	0.72	0.82	0.75	0.85	0.85

Continued on next page

Table D.5 – continued from previous page

Language	**Precision**											
	low				**medium**				**high**			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
Average	0.72	0.67	0.59	0.63	0.79	0.76	0.83	0.84	0.82	0.79	0.88	0.88

Table D.6: Recall at morphological analysis task in each language for morpheme-based system (**M**), holistic approach(**H**), our neural approaches: (**N-`sml`**) and (**N-`fcs`**).

Language	**Recall**											
	low				**medium**				**high**			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
adyghe	0.89	0.98	0.94	0.97	0.96	0.98	0.97	0.98	0.96	0.97	0.99	0.99
albanian	0.57	0.72	0.48	0.46	0.66	0.80	0.89	0.86	0.74	0.80	0.94	0.94
arabic	0.57	0.78	0.26	0.47	0.66	0.79	0.75	0.79	0.69	0.80	0.88	0.89
armenian	0.70	0.83	0.49	0.54	0.79	0.91	0.88	0.87	0.84	0.91	0.93	0.93
asturian	0.80	0.90	0.54	0.73	0.87	0.92	0.88	0.89	0.87	0.90	0.90	0.90
azeri	0.73	0.90	0.74	0.78	0.88	0.94	0.95	0.94	0.92	0.94	0.96	0.96
bashkir	0.87	0.94	0.89	0.91	0.92	0.94	0.94	0.94	0.93	0.94	0.95	0.95
basque	0.67	0.77	0.53	0.51	0.76	0.87	0.96	0.97	0.83	0.89	0.98	0.98
belarusian	0.64	0.78	0.41	0.38	0.73	0.84	0.75	0.71	0.75	0.78	0.81	0.82
bengali	0.85	0.92	0.64	0.66	0.91	0.95	0.91	0.91	0.89	0.91	0.91	0.92
breton	0.91	0.95	0.78	0.82	0.92	0.95	0.96	0.94	0.93	0.94	0.95	0.96
bulgarian	0.73	0.82	0.33	0.50	0.82	0.89	0.86	0.86	0.87	0.91	0.92	0.91
catalan	0.84	0.92	0.55	0.70	0.91	0.95	0.92	0.92	0.92	0.95	0.93	0.93
classical-syriac	0.86	0.93	0.66	0.78	0.83	0.85	0.87	0.88	0.83	0.85	0.88	0.89
cornish	0.75	0.84	0.64	0.60	0.80	0.90	0.90	0.84				
crimean-tatar	0.86	0.94	0.83	0.90	0.93	0.95	0.93	0.92	0.93	0.94	0.97	0.94
czech	0.58	0.75	0.35	0.47	0.70	0.83	0.71	0.69	0.73	0.82	0.76	0.77
danish	0.84	0.92	0.85	0.76	0.88	0.95	0.91	0.91	0.92	0.95	0.94	0.95
estonian	0.75	0.85	0.55	0.63	0.83	0.91	0.90	0.90	0.85	0.90	0.95	0.95

Continued on next page

Table D.6 – continued from previous page

Language	Recall											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
faroese	0.60	0.75	0.47	0.48	0.69	0.81	0.69	0.70	0.76	0.85	0.80	0.79
finnish	0.49	0.74	0.54	0.50	0.76	0.88	0.82	0.86	0.82	0.91	0.93	0.93
friulian	0.81	0.90	0.65	0.53	0.90	0.94	0.89	0.88	0.85	0.85	0.91	0.89
galician	0.74	0.85	0.49	0.67	0.84	0.91	0.85	0.86	0.85	0.89	0.88	0.89
georgian	0.85	0.94	0.80	0.69	0.92	0.95	0.93	0.93	0.94	0.96	0.95	0.96
greek	0.41	0.73	0.38	0.37	0.45	0.84	0.71	0.75	0.53	0.85	0.83	0.83
greenlandic	0.89	0.94	0.85	0.91	0.91	0.94	0.91	0.93				
haida	0.89	0.95	0.50	0.71	0.96	0.98	0.97	0.99	0.97	0.98	0.98	0.99
hebrew	0.60	0.78	0.43	0.50	0.71	0.82	0.81	0.84	0.72	0.83	0.91	0.89
hindi	0.80	0.88	0.76	0.79	0.87	0.95	0.90	0.89	0.87	0.92	0.90	0.91
hungarian	0.75	0.85	0.64	0.59	0.87	0.93	0.88	0.87	0.93	0.96	0.95	0.94
icelandic	0.69	0.87	0.52	0.55	0.74	0.86	0.75	0.74	0.78	0.86	0.81	0.80
ingrian	0.90	0.95	0.88	0.85	0.95	0.96	0.95	0.96				
irish	0.63	0.69	0.60	0.48	0.63	0.80	0.79	0.80	0.68	0.77	0.86	0.86
italian	0.83	0.89	0.50	0.63	0.90	0.95	0.92	0.89	0.91	0.95	0.93	0.92
kabardian	0.94	0.97	0.97	0.96	0.94	0.97	0.99	0.99	0.97	0.97	0.99	0.98
karelian	0.83	0.89	0.55	0.89	0.87	0.89	0.91	0.91				
kashubian	0.79	0.89	0.72	0.81	0.79	0.82	0.78	0.84				
kazakh	0.86	0.94	0.89	0.95	0.91	0.95	0.94	0.96				
khakas	0.83	0.97	0.91	0.92	0.97	0.99	0.99	0.99				
khaling	0.63	0.81	0.52	0.72	0.74	0.83	0.81	0.84	0.74	0.83	0.86	0.87

Continued on next page

Table D.6 – continued from previous page

Language	Recall											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
kurmanji	0.82	0.92	0.47	0.65	0.90	0.95	0.91	0.92	0.93	0.96	0.95	0.96
ladin	0.76	0.88	0.52	0.67	0.82	0.86	0.85	0.82	0.78	0.79	0.85	0.85
latin	0.54	0.72	0.48	0.35	0.74	0.87	0.69	0.71	0.79	0.88	0.80	0.79
latvian	0.55	0.76	0.39	0.40	0.68	0.82	0.67	0.66	0.74	0.83	0.77	0.80
lithuanian	0.58	0.71	0.28	0.28	0.79	0.89	0.78	0.79	0.85	0.90	0.91	0.89
livonian	0.71	0.88	0.55	0.54	0.85	0.88	0.91	0.90	0.86	0.89	0.93	0.93
lower-sorbian	0.64	0.82	0.38	0.35	0.71	0.84	0.69	0.73	0.70	0.75	0.78	0.76
macedonian	0.78	0.90	0.51	0.23	0.86	0.93	0.86	0.85	0.90	0.94	0.90	0.91
maltese	0.78	0.89	0.77	0.85	0.79	0.86	0.93	0.93	0.80	0.87	0.92	0.92
mapudungun	0.92	0.94	0.86	0.89	0.93	0.97	0.97	0.98				
middle-french	0.87	0.93	0.53	0.68	0.89	0.94	0.92	0.92	0.89	0.92	0.93	0.93
middle-high-german	0.64	0.72	0.57	0.70	0.72	0.71	0.80	0.80				
murrinhpatha	0.72	0.79	0.53	0.52	0.75	0.85	0.87	0.87				
navajo	0.62	0.77	0.55	0.61	0.81	0.90	0.83	0.84	0.85	0.90	0.94	0.94
neapolitan	0.84	0.93	0.75	0.80	0.90	0.92	0.92	0.93	0.90	0.91	0.94	0.93
norman	0.72	0.84	0.68	0.79	0.78	0.86	0.51	0.78				
northern-sami	0.70	0.82	0.36	0.56	0.81	0.91	0.84	0.85	0.86	0.91	0.87	0.88
norwegian-bokmaal	0.80	0.91	0.68	0.71	0.86	0.92	0.85	0.83	0.88	0.93	0.90	0.90
norwegian-nynorsk	0.80	0.89	0.73	0.57	0.82	0.90	0.80	0.81	0.90	0.95	0.92	0.91
occitan	0.84	0.91	0.74	0.85	0.93	0.96	0.93	0.94	0.92	0.92	0.95	0.96
old-armenian	0.56	0.76	0.39	0.46	0.73	0.86	0.73	0.75	0.77	0.86	0.78	0.80

Continued on next page

Table D.6 – continued from previous page

Language	Recall											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
old-church-slavonic	0.69	0.83	0.58	0.57	0.70	0.72	0.72	0.69	0.67	0.69	0.69	0.70
old-french	0.80	0.88	0.42	0.67	0.88	0.93	0.89	0.90	0.90	0.94	0.92	0.91
old-irish	0.60	0.74	0.54	0.58	0.72	0.81	0.77	0.73				
old-saxon	0.55	0.78	0.39	0.35	0.70	0.81	0.73	0.72	0.77	0.81	0.83	0.83
pashto	0.61	0.86	0.51	0.51	0.78	0.87	0.82	0.82	0.78	0.81	0.85	0.83
persian	0.55	0.88	0.63	0.78	0.59	0.91	0.92	0.93	0.71	0.92	0.95	0.94
portuguese	0.81	0.88	0.68	0.69	0.87	0.93	0.87	0.87	0.88	0.94	0.89	0.89
quechua	0.69	0.76	0.58	0.53	0.88	0.92	0.90	0.90	0.91	0.94	0.93	0.93
romanian	0.58	0.75	0.51	0.46	0.71	0.84	0.80	0.79	0.76	0.84	0.86	0.85
sanskrit	0.60	0.81	0.33	0.35	0.62	0.81	0.72	0.66	0.68	0.77	0.80	0.79
scottish-gaelic	0.56	0.58	0.54	0.55	0.57	0.62	0.57	0.67				
serbo-croatian	0.55	0.72	0.41	0.41	0.67	0.81	0.69	0.70	0.69	0.82	0.75	0.76
slovak	0.69	0.88	0.59	0.49	0.75	0.87	0.74	0.75	0.73	0.76	0.80	0.80
slovene	0.56	0.79	0.44	0.33	0.64	0.80	0.63	0.62	0.65	0.74	0.69	0.68
sorani	0.69	0.82	0.48	0.61	0.80	0.90	0.91	0.91	0.83	0.90	0.99	0.98
spanish	0.81	0.93	0.66	0.80	0.89	0.94	0.92	0.93	0.92	0.95	0.94	0.95
swahili	0.78	0.85	0.61	0.75	0.86	0.94	0.88	0.92	0.87	0.87	0.92	0.91
swedish	0.70	0.86	0.61	0.48	0.76	0.86	0.78	0.76	0.82	0.88	0.83	0.84
tatar	0.90	0.95	0.77	0.82	0.97	0.98	0.94	0.97	0.96	0.96	0.98	0.96
telugu	0.88	0.89	0.85	0.91								
tibetan	0.56	0.57	0.62	0.63	0.53	0.55	0.56	0.57				

Continued on next page

Table D.6 – continued from previous page

Language	Recall											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
turkish	0.62	0.74	0.45	0.59	0.83	0.92	0.90	0.90	0.89	0.95	0.93	0.92
turkmen	0.87	0.99	0.89	0.95	0.96	1.00	0.98	0.99				
ukrainian	0.65	0.76	0.54	0.50	0.77	0.86	0.77	0.77	0.77	0.80	0.81	0.82
urdu	0.73	0.85	0.69	0.75	0.81	0.91	0.84	0.85	0.80	0.82	0.88	0.88
uzbek	0.86	0.92	0.78	0.96	0.89	0.94	0.94	0.95	0.89	0.94	0.95	0.95
venetian	0.78	0.90	0.51	0.53	0.79	0.88	0.81	0.82	0.77	0.80	0.85	0.83
votic	0.87	0.92	0.83	0.90	0.94	0.97	0.97	0.98	0.94	0.97	0.97	0.97
welsh	0.86	0.96	0.76	0.83	0.91	0.94	0.93	0.95	0.89	0.90	0.93	0.95
west-frisian	0.66	0.80	0.58	0.67	0.66	0.67	0.72	0.78	0.67	0.70	0.70	0.75
yiddish	0.55	0.71	0.53	0.47	0.59	0.73	0.62	0.61	0.54	0.60	0.69	0.73
zulu	0.56	0.69	0.41	0.34	0.77	0.88	0.81	0.81	0.80	0.86	0.89	0.89
dutch	0.58	0.81	0.46	0.55	0.64	0.80	0.61	0.67	0.64	0.74	0.70	0.74
english	0.95	0.98	0.95	0.96	0.95	0.97	0.98	0.92	0.95	0.97	0.99	0.99
french	0.81	0.92	0.65	0.66	0.86	0.94	0.89	0.90	0.89	0.94	0.90	0.91
german	0.60	0.77	0.56	0.53	0.61	0.77	0.71	0.73	0.67	0.78	0.76	0.76
kannada	0.73	0.90	0.42	0.55	0.85	0.96	0.90	0.91	0.90	0.96	0.94	0.92
middle-low-german	0.51	0.77	0.45	0.55	0.63	0.63	0.76	0.79				
north-frisian	0.69	0.81	0.69	0.72	0.70	0.79	0.78	0.81	0.69	0.83	0.78	0.79
old-english	0.54	0.74	0.26	0.40	0.62	0.77	0.63	0.65	0.67	0.74	0.72	0.73
polish	0.60	0.77	0.38	0.34	0.74	0.85	0.74	0.70	0.78	0.86	0.82	0.81
russian	0.68	0.78	0.41	0.31	0.77	0.88	0.77	0.73	0.82	0.90	0.85	0.84

Continued on next page

Table D.6 – continued from previous page

Language	**Recall**											
	low				**medium**				**high**			
	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`	M	H	N-`sml`	N-`fcs`
Average	0.72	0.84	0.59	0.63	0.80	0.88	0.83	0.84	0.82	0.87	0.88	0.88

Table D.7: F1 score at morphological analysis task in each language for morpheme-based system (**M**), holistic approach(**H**), our neural approaches: (**N-sml**) and (**N-fcs**).

Language	**F1 score**											
	low				**medium**				**high**			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
adyghe	0.89	0.87	0.94	0.97	0.96	0.94	0.97	0.98	0.96	0.94	0.99	0.99
albanian	0.57	0.56	0.48	0.46	0.66	0.68	0.89	0.86	0.74	0.72	0.94	0.94
arabic	0.57	0.57	0.27	0.47	0.66	0.64	0.75	0.79	0.69	0.68	0.88	0.89
armenian	0.69	0.71	0.48	0.55	0.78	0.78	0.88	0.87	0.84	0.83	0.93	0.93
asturian	0.79	0.79	0.55	0.71	0.87	0.86	0.88	0.89	0.87	0.87	0.90	0.90
azeri	0.73	0.72	0.73	0.79	0.88	0.87	0.95	0.94	0.92	0.91	0.97	0.96
bashkir	0.88	0.87	0.88	0.91	0.92	0.91	0.94	0.94	0.93	0.92	0.95	0.95
basque	0.65	0.65	0.52	0.49	0.75	0.75	0.95	0.96	0.83	0.83	0.98	0.98
belarusian	0.64	0.64	0.42	0.38	0.73	0.73	0.75	0.71	0.74	0.74	0.81	0.82
bengali	0.83	0.82	0.64	0.65	0.90	0.88	0.91	0.91	0.89	0.89	0.90	0.92
breton	0.90	0.89	0.79	0.83	0.92	0.93	0.95	0.94	0.93	0.93	0.95	0.96
bulgarian	0.73	0.71	0.35	0.50	0.81	0.79	0.85	0.85	0.87	0.86	0.92	0.91
catalan	0.84	0.82	0.54	0.68	0.91	0.91	0.92	0.92	0.92	0.92	0.93	0.93
classical-syriac	0.86	0.82	0.67	0.78	0.83	0.80	0.87	0.88	0.83	0.81	0.88	0.89
cornish	0.75	0.72	0.61	0.60	0.81	0.80	0.90	0.85				
crimean-tatar	0.86	0.85	0.83	0.90	0.93	0.88	0.93	0.92	0.93	0.91	0.97	0.94
czech	0.57	0.58	0.37	0.47	0.70	0.69	0.71	0.69	0.73	0.72	0.76	0.77
danish	0.84	0.81	0.85	0.76	0.88	0.88	0.91	0.91	0.92	0.91	0.94	0.95
estonian	0.74	0.74	0.55	0.62	0.83	0.84	0.90	0.90	0.85	0.85	0.95	0.95

Continued on next page

Table D.7 – continued from previous page

Language	F1 score											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
faroese	0.60	0.63	0.47	0.49	0.69	0.69	0.69	0.70	0.76	0.75	0.80	0.79
finnish	0.48	0.62	0.54	0.49	0.76	0.76	0.82	0.86	0.82	0.82	0.93	0.93
friulian	0.81	0.83	0.64	0.50	0.91	0.90	0.89	0.88	0.85	0.85	0.91	0.89
galician	0.73	0.72	0.47	0.66	0.84	0.83	0.85	0.85	0.85	0.85	0.88	0.89
georgian	0.85	0.85	0.80	0.69	0.91	0.90	0.93	0.93	0.94	0.92	0.95	0.96
greek	0.42	0.59	0.40	0.39	0.45	0.70	0.71	0.75	0.53	0.75	0.83	0.83
greenlandic	0.88	0.92	0.85	0.91	0.91	0.92	0.91	0.93				
haida	0.84	0.82	0.51	0.72	0.93	0.92	0.97	0.99	0.96	0.96	0.98	0.99
hebrew	0.60	0.59	0.41	0.48	0.70	0.68	0.80	0.84	0.72	0.71	0.91	0.89
hindi	0.79	0.78	0.75	0.78	0.87	0.86	0.90	0.89	0.87	0.87	0.90	0.91
hungarian	0.75	0.75	0.64	0.59	0.87	0.84	0.88	0.87	0.93	0.91	0.95	0.94
icelandic	0.69	0.70	0.52	0.55	0.74	0.74	0.75	0.74	0.78	0.78	0.81	0.80
ingrian	0.90	0.87	0.88	0.85	0.95	0.91	0.95	0.96				
irish	0.62	0.55	0.60	0.49	0.63	0.65	0.79	0.79	0.68	0.67	0.85	0.86
italian	0.82	0.80	0.50	0.62	0.90	0.90	0.91	0.89	0.91	0.92	0.93	0.92
kabardian	0.94	0.90	0.96	0.96	0.93	0.94	0.99	0.99	0.96	0.95	0.99	0.98
karelian	0.83	0.83	0.55	0.89	0.87	0.87	0.91	0.91				
kashubian	0.79	0.82	0.72	0.81	0.79	0.80	0.78	0.84				
kazakh	0.85	0.89	0.89	0.95	0.91	0.92	0.94	0.96				
khakas	0.83	0.89	0.91	0.92	0.96	0.98	0.99	0.99				
khaling	0.63	0.67	0.53	0.71	0.74	0.73	0.80	0.84	0.74	0.74	0.86	0.86

Continued on next page

Table D.7 – continued from previous page

Language	F1 score											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
kurmanji	0.83	0.81	0.51	0.66	0.90	0.87	0.92	0.93	0.92	0.90	0.95	0.96
ladin	0.76	0.77	0.51	0.67	0.82	0.80	0.85	0.82	0.78	0.78	0.85	0.85
latin	0.54	0.56	0.48	0.35	0.73	0.75	0.68	0.71	0.79	0.79	0.80	0.79
latvian	0.54	0.55	0.39	0.40	0.67	0.68	0.67	0.66	0.74	0.73	0.77	0.80
lithuanian	0.57	0.56	0.29	0.29	0.79	0.77	0.78	0.78	0.84	0.83	0.91	0.89
livonian	0.71	0.68	0.54	0.53	0.83	0.80	0.90	0.89	0.84	0.85	0.92	0.93
lower-sorbian	0.64	0.65	0.39	0.35	0.71	0.72	0.69	0.73	0.70	0.70	0.78	0.76
macedonian	0.76	0.75	0.51	0.28	0.86	0.84	0.86	0.85	0.90	0.88	0.90	0.91
maltese	0.79	0.79	0.78	0.85	0.79	0.78	0.93	0.93	0.81	0.78	0.93	0.92
mapudungun	0.92	0.90	0.86	0.89	0.93	0.95	0.97	0.98				
middle-french	0.86	0.82	0.51	0.67	0.89	0.89	0.91	0.91	0.89	0.90	0.93	0.93
middle-high-german	0.62	0.61	0.56	0.68	0.70	0.67	0.79	0.78				
murrinhpatha	0.70	0.66	0.54	0.54	0.73	0.73	0.86	0.85				
navajo	0.62	0.63	0.55	0.61	0.81	0.80	0.83	0.84	0.85	0.84	0.94	0.94
neapolitan	0.83	0.83	0.74	0.79	0.89	0.89	0.91	0.92	0.89	0.89	0.94	0.93
norman	0.71	0.75	0.67	0.78	0.77	0.80	0.49	0.77				
northern-sami	0.70	0.73	0.37	0.56	0.81	0.83	0.84	0.85	0.86	0.86	0.87	0.88
norwegian-bokmaal	0.79	0.80	0.68	0.71	0.86	0.86	0.85	0.83	0.88	0.88	0.90	0.90
norwegian-nynorsk	0.80	0.79	0.73	0.57	0.82	0.82	0.80	0.81	0.90	0.90	0.91	0.91
occitan	0.84	0.84	0.73	0.84	0.93	0.93	0.93	0.94	0.92	0.92	0.95	0.96
old-armenian	0.56	0.58	0.40	0.46	0.72	0.71	0.73	0.75	0.76	0.75	0.78	0.80

Continued on next page

Table D.7 – continued from previous page

Language	F1 score											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
old-church-slavonic	0.69	0.70	0.58	0.57	0.70	0.69	0.72	0.69	0.67	0.66	0.69	0.70
old-french	0.80	0.79	0.45	0.66	0.88	0.86	0.89	0.89	0.90	0.89	0.91	0.91
old-irish	0.60	0.58	0.53	0.57	0.71	0.66	0.75	0.72				
old-saxon	0.55	0.62	0.37	0.36	0.69	0.70	0.72	0.72	0.77	0.76	0.83	0.83
pashto	0.61	0.62	0.51	0.50	0.78	0.74	0.83	0.82	0.78	0.73	0.85	0.83
persian	0.54	0.76	0.63	0.74	0.58	0.84	0.92	0.92	0.70	0.87	0.94	0.94
portuguese	0.81	0.78	0.65	0.66	0.86	0.87	0.87	0.87	0.88	0.88	0.88	0.89
quechua	0.68	0.65	0.57	0.53	0.87	0.86	0.90	0.90	0.90	0.88	0.92	0.93
romanian	0.58	0.63	0.50	0.46	0.71	0.70	0.79	0.79	0.76	0.75	0.86	0.84
sanskrit	0.59	0.58	0.34	0.36	0.62	0.62	0.72	0.67	0.68	0.68	0.80	0.79
scottish-gaelic	0.55	0.48	0.53	0.53	0.56	0.57	0.57	0.67				
serbo-croatian	0.54	0.54	0.39	0.40	0.67	0.66	0.69	0.70	0.69	0.69	0.75	0.76
slovak	0.69	0.67	0.58	0.49	0.75	0.75	0.74	0.74	0.73	0.71	0.80	0.80
slovene	0.55	0.58	0.44	0.34	0.64	0.65	0.63	0.62	0.65	0.66	0.69	0.68
sorani	0.68	0.69	0.49	0.62	0.80	0.80	0.90	0.91	0.83	0.84	0.99	0.98
spanish	0.81	0.79	0.66	0.80	0.89	0.89	0.92	0.92	0.92	0.91	0.94	0.94
swahili	0.76	0.73	0.60	0.74	0.86	0.86	0.88	0.92	0.87	0.87	0.92	0.91
swedish	0.69	0.71	0.61	0.48	0.76	0.76	0.78	0.76	0.82	0.81	0.83	0.84
tatar	0.90	0.88	0.77	0.82	0.97	0.92	0.94	0.97	0.96	0.93	0.98	0.96
telugu	0.88	0.86	0.85	0.90								
tibetan	0.56	0.53	0.62	0.63	0.53	0.53	0.56	0.57				

Continued on next page

Table D.7 – continued from previous page

Language	F1 score											
	low				medium				high			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
turkish	0.61	0.61	0.46	0.59	0.82	0.83	0.89	0.89	0.89	0.89	0.93	0.92
turkmen	0.86	0.89	0.89	0.95	0.96	0.97	0.98	0.99				
ukrainian	0.65	0.64	0.54	0.50	0.77	0.75	0.77	0.77	0.76	0.75	0.81	0.82
urdu	0.73	0.72	0.68	0.75	0.81	0.81	0.84	0.85	0.80	0.81	0.88	0.88
uzbek	0.85	0.83	0.78	0.95	0.87	0.90	0.94	0.95	0.88	0.90	0.95	0.95
venetian	0.77	0.77	0.50	0.49	0.79	0.80	0.81	0.82	0.77	0.77	0.84	0.83
votic	0.87	0.86	0.83	0.90	0.94	0.94	0.97	0.98	0.94	0.94	0.97	0.97
welsh	0.85	0.87	0.74	0.83	0.91	0.90	0.92	0.95	0.89	0.89	0.92	0.95
west-frisian	0.64	0.66	0.53	0.63	0.65	0.65	0.70	0.76	0.65	0.67	0.66	0.73
yiddish	0.55	0.55	0.53	0.46	0.59	0.62	0.61	0.62	0.55	0.57	0.69	0.73
zulu	0.56	0.56	0.43	0.36	0.77	0.78	0.80	0.81	0.80	0.80	0.89	0.89
dutch	0.57	0.58	0.44	0.53	0.63	0.64	0.58	0.65	0.63	0.64	0.70	0.71
english	0.92	0.92	0.93	0.92	0.94	0.94	0.94	0.94	0.95	0.94	0.95	0.95
french	0.80	0.79	0.65	0.65	0.87	0.87	0.89	0.90	0.89	0.89	0.90	0.91
german	0.60	0.57	0.55	0.52	0.61	0.60	0.70	0.73	0.67	0.66	0.75	0.76
kannada	0.73	0.75	0.42	0.56	0.85	0.86	0.89	0.90	0.89	0.89	0.94	0.92
middle-low-german	0.50	0.58	0.45	0.51	0.62	0.60	0.75	0.77				
north-frisian	0.69	0.70	0.68	0.71	0.70	0.74	0.77	0.80	0.69	0.72	0.77	0.78
old-english	0.54	0.58	0.26	0.39	0.61	0.60	0.61	0.64	0.66	0.67	0.71	0.72
polish	0.60	0.61	0.39	0.35	0.74	0.72	0.74	0.70	0.78	0.77	0.81	0.81
russian	0.67	0.65	0.41	0.33	0.77	0.76	0.77	0.72	0.81	0.80	0.85	0.84

Continued on next page

Table D.7 – continued from previous page

Language	**F1 score**											
	low				**medium**				**high**			
	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs	M	H	N-sml	N-fcs
Average	0.72	0.72	0.59	0.63	0.79	0.80	0.83	0.84	0.82	0.82	0.88	0.88

Table D.8: Morphological complexity (C_{WALS}) for all dataset sizes computed using the Formula (3.8). f_i stands for the total number of unique MSF on a particular MSF group exists in the language. This number is then normalised against the maximum number of MSF value in that particular MSF group. n is the number of the feature (MSF group).

Language	**low**			**medium**			**high**		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
adyghe	9	5	0.2353	9	5	0.2353	9	5	0.2353
albanian	26	9	0.2874	26	9	0.2874	26	9	0.2874
arabic	26	13	0.2246	28	13	0.2327	28	13	0.2327
armenian	28	11	0.2927	30	12	0.2791	30	12	0.2791
asturian	22	9	0.3839	25	9	0.4342	25	9	0.4342
azeri	20	8	0.1883	22	8	0.2178	22	8	0.2178
bashkir	10	4	0.1811	10	4	0.1811	10	4	0.1811
basque	7	4	0.1719	7	4	0.1719	7	4	0.1719
belarusian	23	8	0.3015	23	8	0.3015	23	8	0.3015
bengali	23	10	0.2625	23	10	0.2625	23	10	0.2625
breton	18	8	0.3011	20	8	0.3439	20	8	0.3439
bulgarian	28	12	0.2634	29	12	0.2678	29	12	0.2678
catalan	19	8	0.3075	20	9	0.3289	20	9	0.3289
classical-syriac	19	6	0.2326	19	6	0.2326	19	6	0.2326
cornish	22	9	0.4068	22	9	0.4068			
crimean-tatar	14	7	0.1860	16	7	0.2113	16	7	0.2113
czech	32	11	0.2745	33	12	0.2933	33	12	0.2933
danish	15	8	0.2300	16	8	0.2456	16	8	0.2456
estonian	29	11	0.2955	29	11	0.2955	29	11	0.2955

Continued on next page

Table D.8 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
faroese	22	9	0.3102	23	9	0.3160	23	9	0.3160
finnish	29	9	0.3073	30	10	0.3265	30	10	0.3265
friulian	17	7	0.2943	20	8	0.3891	20	8	0.3891
galician	21	9	0.3781	25	10	0.4655	25	10	0.4655
georgian	21	8	0.3127	27	8	0.3725	27	8	0.3725
greek	24	8	0.3052	29	10	0.3594	29	10	0.3594
greenlandic	11	3	0.1644	11	3	0.1644			
haida	1	1	0.0526	1	1	0.0526	1	1	0.0526
hebrew	20	9	0.2703	20	9	0.2703	20	9	0.2703
hindi	16	7	0.2804	23	10	0.3812	23	10	0.3812
hungarian	23	8	0.2895	25	8	0.3117	25	8	0.3117
icelandic	18	7	0.2417	21	8	0.2871	21	8	0.2871
ingrian	9	3	0.1581	9	3	0.1581			
irish	22	8	0.2370	28	10	0.2744	28	10	0.2744
italian	18	7	0.3018	19	8	0.3266	19	8	0.3266
kabardian	9	5	0.2353	9	5	0.2353	9	5	0.2353
karelian	13	3	0.1898	14	3	0.1978			
kashubian	10	3	0.1564	10	3	0.1564			
kazakh	8	3	0.1406	8	3	0.1406			
khakas	11	3	0.1644	11	3	0.1644			
khaling	17	7	0.3801	18	8	0.3951	18	8	0.3951
kurmanji	28	10	0.4397	29	10	0.4450	30	10	0.4503

Continued on next page

Table D.8 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
ladin	16	8	0.3515	17	8	0.3580	17	8	0.3580
latin	27	9	0.2803	29	10	0.3075	29	10	0.3075
latvian	25	10	0.2923	29	11	0.3684	29	11	0.3684
lithuanian	28	10	0.2952	30	11	0.3593	30	11	0.3593
livonian	28	12	0.3195	30	12	0.3655	30	12	0.3655
lower-sorbian	21	7	0.2731	23	8	0.3015	23	8	0.3015
macedonian	25	11	0.2595	26	11	0.2725	26	11	0.2725
maltese	14	8	0.3124	15	8	0.3190	15	8	0.3190
mapudungun	10	4	0.2297	10	4	0.2297			
middle-french	19	7	0.3732	20	8	0.3891	20	8	0.3891
middle-high-german	17	7	0.2458	18	7	0.2492			
murrinhpatha	8	5	0.1627	8	5	0.1627			
navajo	17	7	0.3292	19	7	0.3415	19	7	0.3415
neapolitan	20	8	0.3891	20	8	0.3891	20	8	0.3891
norman	24	10	0.4512	24	10	0.4512			
northern-sami	21	6	0.2449	21	6	0.2449	21	6	0.2449
norwegian-bokmaal	13	7	0.2305	13	7	0.2305	19	10	0.2638
norwegian-nynorsk	15	9	0.2433	15	9	0.2433	20	10	0.2715
occitan	18	7	0.3018	18	7	0.3018	20	8	0.3891
old-armenian	30	11	0.4301	32	12	0.4091	34	13	0.4161
old-church-slavonic	11	3	0.1981	11	3	0.1981	11	3	0.1981
old-french	19	8	0.3825	25	10	0.3789	25	10	0.3789

Continued on next page

Table D.8 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
old-irish	30	10	0.3653	32	10	0.3729			
old-saxon	26	10	0.3797	26	10	0.3797	26	10	0.3797
pashto	19	8	0.3562	19	8	0.3562	19	8	0.3562
persian	17	7	0.2596	18	8	0.2897	18	8	0.2897
portuguese	22	9	0.4003	22	9	0.4003	22	9	0.4003
quechua	36	10	0.2835	38	10	0.3446	38	10	0.3446
romanian	27	11	0.3915	27	11	0.3915	27	11	0.3915
sanskrit	15	4	0.3117	15	4	0.3117	15	4	0.3117
scottish-gaelic	17	7	0.2121	17	7	0.2121			
serbo-croatian	34	13	0.3177	36	13	0.3754	36	13	0.3754
slovak	15	5	0.3196	15	5	0.3196	15	5	0.3196
slovene	24	8	0.3164	32	11	0.3222	32	11	0.3222
sorani	29	14	0.2739	34	15	0.3434	34	15	0.3434
spanish	23	10	0.4012	23	10	0.4012	23	10	0.4012
swahili	18	9	0.3726	19	9	0.3784	19	9	0.3784
swedish	21	9	0.2771	22	10	0.2694	22	10	0.2694
tatar	15	7	0.2038	16	7	0.2113	16	7	0.2113
telugu	14	7	0.2268						
tibetan	5	3	0.1601	5	3	0.1601			
turkish	30	12	0.3546	30	12	0.3546	30	12	0.3546
turkmen	8	3	0.1406	8	3	0.1406			
ukrainian	25	8	0.3206	28	10	0.3118	29	10	0.3195

Continued on next page

Table D.8 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
urdu	26	11	0.3508	27	11	0.3578	27	11	0.3578
uzbek	14	4	0.1554	14	4	0.1554	14	4	0.1554
venetian	19	8	0.3712	19	8	0.3712	19	8	0.3712
votic	10	3	0.1564	10	3	0.1564	10	3	0.1564
welsh	18	7	0.2428	18	7	0.2428	18	7	0.2428
west-frisian	15	8	0.2424	15	8	0.2424	15	8	0.2424
yiddish	21	10	0.3619	21	10	0.3619	21	10	0.3619
zulu	17	7	0.3473	18	8	0.3664	18	8	0.3664
dutch	19	9	0.3055	19	9	0.3055	19	9	0.3055
english	7	5	0.2183	7	5	0.2183	7	5	0.2183
french	19	8	0.3266	19	8	0.3266	19	8	0.3266
german	17	6	0.2074	18	7	0.2492	18	7	0.2492
kannada	27	10	0.4493	28	10	0.4546	28	10	0.4546
middle-low-german	25	9	0.3833	25	9	0.3833			
north-frisian	18	8	0.3940	19	8	0.4006	19	8	0.4006
old-english	25	9	0.3801	25	9	0.3801	25	9	0.3801
polish	27	9	0.2716	32	10	0.3550	32	10	0.3550
russian	27	10	0.2921	29	11	0.3337	29	11	0.3337

Table D.9: Estimated morphological complexity (C_{WALS}) for all dataset sizes computed using the Formula (3.8). $\sum_{i=1}^{n} f_i$ stands for the total number of unique MSF, while n is the length of the longest MSD.

Language	**low**			**medium**			**high**		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
adyghe	10	4	2.50	10	4	2.50	10	5	2.00
albanian	27	6	4.50	27	6	4.50	27	6	4.50
arabic	27	8	3.38	29	8	3.63	29	8	3.63
armenian	34	7	4.86	37	7	5.29	38	7	5.43
asturian	22	7	3.14	26	7	3.71	30	7	4.29
azeri	21	5	4.20	23	5	4.60	23	5	4.60
bashkir	11	4	2.75	11	4	2.75	11	4	2.75
basque	32	11	2.91	32	12	2.67	32	12	2.67
belarusian	23	5	4.60	23	5	4.60	23	5	4.60
bengali	26	5	5.20	26	5	5.20	26	5	5.20
breton	18	8	2.25	20	8	2.50	20	9	2.22
bulgarian	28	7	4.00	29	7	4.14	29	7	4.14
catalan	21	6	3.50	22	6	3.67	22	6	3.67
classical-syriac	19	4	4.75	19	4	4.75	19	4	4.75
cornish	24	8	3.00	24	8	3.00			
crimean-tatar	15	4	3.75	17	4	4.25	17	4	4.25
czech	32	7	4.57	33	7	4.71	33	7	4.71
danish	15	4	3.75	16	4	4.00	16	4	4.00
estonian	35	8	4.38	36	8	4.50	36	8	4.50
faroese	24	5	4.80	26	5	5.20	26	5	5.20

Continued on next page

Table D.9 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
finnish	35	8	4.38	37	8	4.63	37	8	4.63
friulian	17	6	2.83	20	6	3.33	20	6	3.33
galician	21	6	3.50	25	6	4.17	25	6	4.17
georgian	22	6	3.67	28	6	4.67	28	6	4.67
greek	24	5	4.80	30	6	5.00	31	6	5.17
greenlandic	11	3	3.67	11	3	3.67			
haida	26	6	4.33	29	7	4.14	30	7	4.29
hebrew	24	5	4.80	24	5	4.80	24	5	4.80
hindi	18	6	3.00	25	6	4.17	25	6	4.17
hungarian	33	6	5.50	35	6	5.83	35	6	5.83
icelandic	18	5	3.60	21	5	4.20	21	5	4.20
ingrian	15	3	5.00	15	3	5.00			
irish	24	5	4.80	33	6	5.50	35	6	5.83
italian	18	6	3.00	19	6	3.17	19	6	3.17
kabardian	10	4	2.50	10	4	2.50	10	4	2.50
karelian	19	3	6.33	20	3	6.67			
kashubian	10	3	3.33	10	3	3.33			
kazakh	10	3	3.33	10	3	3.33			
khakas	11	3	3.67	11	3	3.67			
khaling	25	9	2.78	26	9	2.89	26	9	2.89
kurmanji	28	10	2.80	29	14	2.07	30	14	2.14
ladin	17	6	2.83	18	7	2.57	19	7	2.71

Continued on next page

Table D.9 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
latin	28	7	4.00	33	7	4.71	33	7	4.71
latvian	29	5	5.80	36	5	7.20	37	5	7.40
lithuanian	32	6	5.33	34	6	5.67	34	6	5.67
livonian	33	10	3.30	37	10	3.70	38	10	3.80
lower-sorbian	21	4	5.25	23	5	4.60	23	5	4.60
macedonian	25	5	5.00	26	6	4.33	26	6	4.33
maltese	16	7	2.29	17	7	2.43	17	7	2.43
mapudungun	10	4	2.50	10	4	2.50			
middle-french	19	7	2.71	20	7	2.86	21	7	3.00
middle-high-german	17	5	3.40	18	5	3.60			
murrinhpatha	11	5	2.20	11	5	2.20			
navajo	24	5	4.80	26	5	5.20	26	5	5.20
neapolitan	20	6	3.33	20	7	2.86	20	7	2.86
norman	24	7	3.43	24	7	3.43			
northern-sami	23	5	4.60	23	5	4.60	23	5	4.60
norwegian-bokmaal	15	3	5.00	15	3	5.00	21	4	5.25
norwegian-nynorsk	17	3	5.67	18	4	4.50	23	4	5.75
occitan	18	6	3.00	18	6	3.00	20	6	3.33
old-armenian	35	8	4.38	38	8	4.75	44	8	5.50
old-church-slavonic	11	3	3.67	11	3	3.67	11	3	3.67
old-french	22	7	3.14	31	7	4.43	34	7	4.86
old-irish	32	8	4.00	36	8	4.50			

Continued on next page

Table D.9 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
old-saxon	26	6	4.33	26	6	4.33	26	6	4.33
pashto	20	7	2.86	20	7	2.86	20	7	2.86
persian	19	6	3.17	21	6	3.50	21	6	3.50
portuguese	22	6	3.67	22	6	3.67	22	6	3.67
quechua	42	5	8.40	46	5	9.20	46	5	9.20
romanian	30	6	5.00	30	6	5.00	30	6	5.00
sanskrit	16	4	4.00	16	4	4.00	16	4	4.00
scottish-gaelic	17	4	4.25	17	4	4.25			
serbo-croatian	34	6	5.67	36	6	6.00	36	6	6.00
slovak	15	5	3.00	15	5	3.00	15	5	3.00
slovene	24	5	4.80	33	5	6.60	33	5	6.60
sorani	29	9	3.22	34	12	2.83	34	12	2.83
spanish	23	6	3.83	23	6	3.83	23	6	3.83
swahili	28	8	3.50	32	8	4.00	32	8	4.00
swedish	22	5	4.40	23	5	4.60	23	5	4.60
tatar	16	4	4.00	17	4	4.25	17	4	4.25
telugu	14	6	2.33						
tibetan	5	2	2.50	5	2	2.50			
turkish	33	8	4.13	37	8	4.63	37	8	4.63
turkmen	9	3	3.00	9	3	3.00			
ukrainian	25	4	6.25	28	5	5.60	29	5	5.80
urdu	28	6	4.67	29	6	4.83	29	6	4.83

Continued on next page

Table D.9 – continued from previous page

Language	low			medium			high		
	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}	$\sum_{i=1}^{n} f_i$	n	C_{WALS}
uzbek	15	4	3.75	15	4	3.75	15	4	3.75
venetian	19	6	3.17	20	7	2.86	21	7	3.00
votic	16	3	5.33	16	3	5.33	16	3	5.33
welsh	18	7	2.57	18	7	2.57	18	7	2.57
west-frisian	15	6	2.50	15	6	2.50	15	6	2.50
yiddish	27	5	5.40	27	5	5.40	27	5	5.40
zulu	43	6	7.17	48	6	8.00	48	6	8.00
dutch	19	5	3.80	19	5	3.80	19	5	3.80
english	7	4	1.75	7	4	1.75	7	4	1.75
french	19	6	3.17	19	6	3.17	19	6	3.17
german	17	5	3.40	18	5	3.60	18	5	3.60
kannada	27	5	5.40	28	5	5.60	28	5	5.60
middle-low-german	25	5	5.00	25	5	5.00			
north-frisian	18	7	2.57	19	7	2.71	19	7	2.71
old-english	25	5	5.00	25	5	5.00	25	5	5.00
polish	27	6	4.50	32	6	5.33	32	6	5.33
russian	27	5	5.40	29	5	5.80	29	5	5.80

Table D.10: Number of productive (P), unproductive ($\neg P$), and total ($|R|$) rule computed on each language using Tolerance Principle (Formula (3.5)).

Language	**low**			**medium**			**high**		
	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$
adyghe	16	5	21	24	0	24	24	6	30
albanian	21	53	74	136	4	140	140	0	140
arabic	17	58	75	127	62	189	79	117	196
armenian	13	68	81	171	34	205	220	0	220
asturian	32	21	53	83	16	99	129	52	181
azeri	22	19	41	62	18	80	79	9	88
bashkir	9	8	17	21	3	24	24	0	24
basque	5	90	95	223	503	726	498	1151	1649
belarusian	23	24	47	8	48	56	1	55	56
bengali	33	15	48	45	13	58	46	12	58
breton	26	22	48	24	62	86	30	78	108
bulgarian	30	29	59	90	5	95	95	0	95
catalan	27	19	46	50	3	53	53	0	53
classical-syriac	18	10	28	38	0	38	38	0	38
cornish	29	28	57	8	97	105			
crimean-tatar	10	3	13	17	0	17	17	0	17
czech	20	45	65	131	30	161	182	6	188
danish	7	3	10	14	0	14	14	0	14
estonian	25	36	61	95	14	109	100	9	109
faroese	23	10	33	37	10	47	38	9	47

Continued on next page

Table D.10 – continued from previous page

Language	low			medium			high		
	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$
finnish	24	46	70	132	57	189	129	68	197
friulian	29	12	41	48	3	51	50	4	54
galician	26	29	55	73	5	78	82	2	84
georgian	19	9	28	38	24	62	82	27	109
greek	21	35	56	97	59	156	113	65	178
greenlandic	11	5	16	7	9	16			
haida	18	62	80	168	8	176	138	41	179
hebrew	26	16	42	20	34	54	22	32	54
hindi	10	78	88	203	6	209	211	0	211
hungarian	25	18	43	74	18	92	91	2	93
icelandic	18	15	33	31	13	44	32	12	44
ingrian	16	9	25	1	42	43			
irish	14	19	33	54	28	82	36	53	89
italian	27	15	42	48	3	51	51	0	51
kabardian	13	3	16	24	0	24	24	0	24
karelian	27	10	37	44	18	62			
kashubian	9	5	14	7	7	14			
kazakh	14	0	14	13	1	14			
khakas	16	0	16	16	0	16			
khaling	13	73	86	235	132	367	45	386	431
kurmanji	14	14	28	50	26	76	92	16	108
ladin	27	16	43	54	10	64	73	20	93
latin	20	49	69	94	57	151	3	148	151

Continued on next page

Table D.10 – continued from previous page

Language	low			medium			high		
	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$
latvian	30	18	48	70	9	79	63	17	80
lithuanian	19	43	62	99	34	133	113	26	139
livonian	24	25	49	84	36	120	44	83	127
lower-sorbian	22	18	40	65	7	72	71	1	72
macedonian	28	19	47	59	17	76	109	2	111
maltese	4	16	20	5	22	27	9	22	31
mapudungun	22	3	25	27	0	27			
middle-french	29	24	53	62	6	68	74	18	92
middle-high-german	25	9	34	21	17	38			
murrinhpatha	25	11	36	2	35	37			
navajo	19	18	37	10	44	54	13	47	60
neapolitan	31	13	44	46	4	50	46	5	51
norman	34	17	51	53	3	56			
northern-sami	28	25	53	19	61	80	12	68	80
norwegian-bokmaal	11	2	13	13	1	14	23	6	29
norwegian-nynorsk	11	3	14	16	5	21	24	9	33
occitan	29	16	45	49	0	49	48	5	53
old-armenian	19	53	72	150	42	192	177	67	244
old-church-slavonic	14	6	20	4	17	21	6	15	21
old-french	28	33	61	79	50	129	153	86	239
old-irish	15	58	73	51	197	248			
old-saxon	20	41	61	85	55	140	0	149	149
pashto	21	37	58	101	13	114	79	39	118

Continued on next page

Table D.10 – continued from previous page

Language	low			medium			high		
	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$
persian	22	53	75	113	23	136	82	54	136
portuguese	28	28	56	75	1	76	76	0	76
quechua	13	71	84	271	125	396	551	2	553
romanian	28	17	45	50	9	59	52	7	59
sanskrit	23	32	55	95	9	104	119	1	120
scottish-gaelic	14	5	19	9	10	19			
serbo-croatian	16	63	79	208	68	276	295	5	300
slovak	18	12	30	30	9	39	38	1	39
slovene	25	28	53	86	11	97	81	18	99
sorani	18	62	80	191	38	229	152	98	250
spanish	29	28	57	67	3	70	69	1	70
swahili	20	56	76	181	11	192	206	1	207
swedish	20	4	24	31	2	33	34	0	34
tatar	8	5	13	17	0	17	17	0	17
telugu	9	8	17						
tibetan	1	3	4	1	3	4			
turkish	24	46	70	154	84	238	276	32	308
turkmen	11	1	12	12	0	12			
ukrainian	11	17	28	44	16	60	55	10	65
urdu	16	66	82	205	9	214	217	0	217
uzbek	28	34	62	84	0	84	84	0	84
venetian	30	18	48	60	3	63	69	26	95
votic	20	5	25	1	25	26	1	25	26

Continued on next page

Table D.10 – continued from previous page

Language	**low**			**medium**			**high**		
	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$	P	$\neg P$	$\|R\|$
welsh	30	23	53	18	45	63	1	62	63
west-frisian	12	8	20	9	11	20	9	11	20
yiddish	19	5	24	25	2	27	28	1	29
zulu	24	47	71	149	41	190	218	11	229
dutch	19	5	24	17	8	25	18	7	25
english	5	0	5	5	0	5	5	0	5
french	29	13	42	46	3	49	48	1	49
german	19	5	24	37	0	37	34	3	37
kannada	26	32	58	68	27	95	65	30	95
middle-low-german	20	17	37	10	42	52			
north-frisian	29	18	47	25	33	58	33	32	65
old-english	24	39	63	21	73	94	0	94	94
polish	17	38	55	92	13	105	100	11	111
russian	24	26	50	58	20	78	92	8	100

Appendix E

Program List

The following is a brief documentation of the programs and scripts we created or modified for the work reported in this thesis.

As of September 2023, these programs and scripts can be found on the GitLab server of the EBMT/NLP lab at the following location:

```
http://133.9.48.111:8082/FAM_Rashel
```

In general, they can be categorized into two groups:

- Tools for the production of analogical grids: `nlg` package
  - This package is also available under the following link: `lepage-lab.ips.waseda.ac.jp/nlg-module`
- Tools for morphological tasks: `balderdash` and `palabras`

References

Aharoni, R., Johnson, M., and Firat, O. (2019). Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (NAACL–2019)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Ahlberg, M., Forsberg, M., and Hulden, M. (2015). Paradigm classification in supervised learning of morphology. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL–2015)*, pages 1024–1029, Denver, Colorado. Association for Computational Linguistics.

Anderson, S. R. (1992). A-morphous morphology. *Cambridge studies in linguistics*, 62:452 pages.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference for Learning Representations (ICLR–15)*, San Diego.

Beesley, K. R. (1998). Consonant spreading in Arabic stems. In *Proceedings of COLING-ACL–1998*, volume I, pages 117–123, Montréal.

Bentz, C., Ruzsics, T., Koplenig, A., and Samardžić, T. (2016). A comparison between morphological complexity measures: Typological data vs. language corpora. In *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)*, pages 142–153, Osaka, Japan. The COLING 2016 Organizing Committee.

Bergmanis, T., Kann, K., Schütze, H., and Goldwater, S. (2017). Training data augmentation for low-resource morphological inflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 31–39, Vancouver. Association for Computational Linguistics.

Botha, J. A. and Blunsom, P. (2014). Compositional Morphology for Word Representations and Language Modelling. In *Proceedings of the 31st International Conference on Machine Learning (ICML–2014)*, Beijing, China.

Bron, C. and Kerbosch, J. (1973). Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577.

Chan, E. (2008). *Structures and distributions in morphology learning*. PhD thesis, University of Pennsylvania.

Chan, K., Kaszefski-Yaschuk, S. P., Saran, C., Marquer, E., and Couceiro, M. (2022). Solving morphological analogies through generation. In Couceiro, M. and Murena, P., editors, *Proceedings of the Workshop on the Interactions between Analogical Reasoning and Machine Learning (International Joint Conference on Artificial Intelligence - European Conference on Artificial Intelligence (IJAI-ECAI 2022)), Vienna, Austria, July 23, 2022*, volume 3174 of *CEUR Workshop Proceedings*, pages 29–39, Vienna, Austria. CEUR-WS.org.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP–2014)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Walther, G., Vylomova, E., McCarthy, A. D., Kann, K., Mielke, S., Nicolai, G., Silfverberg, M., Yarowsky, D., Eisner, J., and Hulden, M. (2018). The CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection. In *Proceedings of the CoNLL SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection*, pages 1–27. Association for Computational Linguistics.

Creutz, M. and Lagus, K. (2007). Unsupervised models for morpheme segmentation and morphology learning. *ACM Trans. Speech Lang. Process.*, 4(1):3:1–3:34.

Dinakaramani, A., Rashel, F., Luthfi, A., and Manurung, R. (2014). Designing an Indonesian part of speech tagset and manually tagged Indonesian corpus. In *Proceedings of 2014 International Conference on Asian Language Processing (IALP–2014)*, pages 66–69, Kuching, Malaysia.

Dryer, M. and Eisner, J. (2011). Discovering morphological paradigms from plain text using a dirichlet process mixture model. In *Proceedings of the*

2011 Conference on Empirical Methods in Natural Language Processing (EMNLP–2011), pages 616–627, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Fam, R. and Lepage, Y. (2016a). An empirical property of the density of paradigm tables. In *10th International collaboration Symposium on Information, Production and Systems (ISIPS–2016)*, page (no pagination), Kitakyushu, Japan.

Fam, R. and Lepage, Y. (2016b). Morphological predictability of unseen words using computational analogy. In *Proceedings of the Computational Analogy Workshop at the 24th International Conference on Case-Based Reasoning (ICCBR-CA–2016)*, pages 51–60, Atlanta, Georgia.

Fam, R. and Lepage, Y. (2017a). A holistic approach at a morphological inflection task. In *Proceedings of the 8th Language and Technology Conference (LTC–2017)*, pages 88–92, Poznań, Poland. Fundacja uniwersytetu im. Adama Mickiewicza.

Fam, R. and Lepage, Y. (2017b). A study of the saturation of analogical grids agnostically extracted from texts. In *Proceedings of the Computational Analogy Workshop at the 25th International Conference on Case-Based Reasoning (ICCBR-CA–2017)*, pages 11–20, Trondheim, Norway.

Fam, R. and Lepage, Y. (2018a). IPS-WASEDA system at CoNLL–SIGMORPHON 2018 shared task on morphological inflection. In *Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection (CoNLL–18)*, pages 33–42, Brussels. Association for Computational Linguistics.

Fam, R. and Lepage, Y. (2018b). Tools for The Production of Analogical Grids and a Resource of N-gram Analogical Grids in 11 Languages. In chair), N. C. C., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Hasida, K., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S., and Tokunaga, T., editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC–2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Fam, R. and Lepage, Y. (2018c). Validating analogically generated Indonesian words using Fisher's exact test. In *Proceedings of the 24rd Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2018)*, pages 312–315, Okayama, Japan.

Fam, R. and Lepage, Y. (2019). A study of analogical grids extracted using feature vectors on varying vocabulary sizes in Indonesian. In *Proceedings of 2019 International Conference on Advanced Computer Science and Information Systems (ICACSIS–19)*, pages 255–260, Bali, Indonesia.

Fam, R. and Lepage, Y. (2021). A study of analogical density in various corpora at various granularity. *Information*, 12(8).

Fam, R. and Lepage, Y. (2022). Organising lexica into analogical grids: A study of a holistic approach for morphological generation under various sizes of data in various languages. *Journal of Experimental & Theoretical Artificial Intelligence*, 0(0):1–26.

Fam, R. and Lepage, Y. (2023a). Investigating parallelograms: Assessing several word embedding spaces against various analogy test sets in several languages using approximation. In *Proceedings of the 10th Language and Technology Conference (LTC–2023)*, pages 68–72, Poznań, Poland. Fundacja uniwersytetu im. Adama Mickiewicza.

Fam, R. and Lepage, Y. (2023b). Investigating parallelograms inside word embedding space using various analogy test sets in various languages. In *Proceedings of the 29th Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2023)*, pages 718–722, Okinawa, Japan.

Fam, R. and Lepage, Y. (2023c). A resource of sentence analogies on the level of form extracted from corpora in various languages. In *Proceedings of the 29th Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2023)*, pages 103–107, Okinawa, Japan.

Fam, R., Lepage, Y., Gojali, S., and Purwarianti, A. (2017a). Indonesian unseen words explained by form, morphology and distributional semantics at the same time. In *Proceedings of the 23rd Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2017)*, pages 178–181, Tsukuba, Japan.

Fam, R., Lepage, Y., Gojali, S., and Purwarianti, A. (2017b). A study of explaining unseen words in Indonesian using analogical clusters. In *Proceedings of the 15th International Conference on Computer Applications (ICCA–2017)*, pages 416–421, Yangon, Myanmar.

Fam, R., Liu, P., and Lepage, Y. (2019). Checking the validity of word forms generated to fill empty cells in analogical grids. In *Proceedings of* *the 25th Annual Meeting of the Japanese Association for Natural Language Processing (NLP–2019)*, pages 530–533, Nagoya, Japan.

Fam, R., Purwarianti, A., and Lepage, Y. (2017c). Plausibility of word forms generated from analogical grids in Indonesian. In *11th International collaboration Symposium on Information, Production and Systems (ISIPS–2017)*, pages 245–247, Kitakyushu, Japan.

Fam, R., Purwarianti, A., and Lepage, Y. (2018). Plausibility of word forms generated from analogical grids in Indonesian. In *Proceedings of the 16th International Conference on Computer Applications (ICCA–2018)*, pages 179–184, Yangon, Myanmar. UCSY.

Fisher, R. A. (1922). On the interpretation of X^2 from contingency tables, and the calculation of P. *Journal of the Royal Statistical Society*, 85(1):87–94.

Ganter, B. and Wille, R. (1999). *Formal Concept Analysis: Mathematical Foundations*. Springer-Verlag Berlin Heidelberg.

Gasser, M. (2009). Semitic morphological analysis and generation using finite state transducers with feature structures. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL–2009)*, pages 309–317, Athens, Greece. Association for Computational Linguistics.

Gasser, M. (2011). HornMorpho: a system for morphological processing of Amharic, Oromo, and Tigrinya. In *Proceedings of Conference for Human Language Technology for Development*, pages 94–99, Alexandria, Egypt.

Gaume, B., Venant, F., and Victorri, B. (2006). Hierarchy in lexical organisation of natural languages. In Pumain, D., editor, *Hierarchy in Natural and Social Sciences*, pages 121–142. Springer Netherlands, Dordrecht.

Gil, D. (2002). From repetition to reduplication in Riau Indonesian. In *Graz reduplication conference*, pages: 2.

Goldsmith, J. (2001). Unsupervised learning of the morphology of a natural language. *Computational Linguistics*, 27:153–198.

Hathout, N. (2008). Acquistion of the morphological structure of the lexicon based on lexical similarity and formal analogy. In *Proceedings of the 3rd Textgraphs workshop on Graph-based Algorithms for Natural Language Processing*, pages 1–8, Manchester, UK. Coling 2008 Organizing Committee.

Hathout, N. (2009). Acquisition of morphological families and derivational series from a machine readable dictionary. *CoRR*, abs/0905.1609.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Irvine, A. and Callison-Burch, C. (2014). Hallucinating phrase translations for low resource MT. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 160–170, Ann Arbor, Michigan. Association for Computational Linguistics.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., Hughes, M., and Dean, J. (2017). Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Kann, K. and Schütze, H. (2016). Med: The LMU system for the SIGMORPHON 2016 shared task on morphological reinflection. In *Proceedings of the 2016 Meeting of SIGMORPHON*, Berlin, Germany. Association for Computational Linguistics.

Kann, K. and Schütze, H. (2017). The LMU system for the CoNLL-SIGMORPHON 2017 Shared Task on universal morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 40–48, Vancouver. Association for Computational Linguistics.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations (ICLR–2015)*, San Diego.

Kirov, C., Cotterell, R., Sylak-Glassman, J., Walther, G., Vylomova, E., Xia, P., Faruqui, M., Mielke, S., McCarthy, A. D., Kübler, S., Yarowsky, D., Eisner, J., and Hulden, M. (2018). UniMorph 2.0: Universal Morphology. In chair), N. C. C., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Hasida, K., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S., and Tokunaga, T., editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC–2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Kurimo, M., Virpioja, S., Turunen, V., and Lagus, K. (2010). Morpho challenge 2005-2010: Evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 87–95, Uppsala, Sweden. Association for Computational Linguistics.

Kuroda, K. (2016). How are inflectional paradigms represented (in the mind)? Formal Concept Analysis meets Czech declensional paradigms. In *Proceedings of the 22th Annual Conference of the Japanese Association for Natural Language Processing*, pages 849–852, Sendai, Japan. The Association for Natural Language Processing.

Kuryłowicz, J. (1961). *L'apophonie en sémitique.* Ossolineum, Wrocław–Warszawa–Kraków.

Langlais, P. and Patry, A. (2007). Translating unknown words by analogical learning. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL–2007)*, pages 877–886.

Langlais, P. and Yvon, F. (2008). Scaling up analogical learning. In *Coling 2008: Companion volume: Posters*, pages 51–54, Manchester, UK. Coling 2008 Organizing Committee.

Lepage, Y. (1998). Solving analogies on words: an algorithm. In *Proceedings of the 17th international conference on Computational linguistics (COLING–1998)*, volume 1, pages 728–734. Association for Computational Linguistics.

Lepage, Y. (2004). Lower and higher estimates of the number of "true analogies" between sentences contained in a large multilingual corpus. In *Proceedings of 20th International Conference on Computational Linguistics (COLING–2004)*, volume 1, pages 736–742, Genève.

Lepage, Y. (2014). Analogies between binary images: Application to Chinese characters. In Prade, H. and Richard, G., editors, *Computational Approaches to Analogical Reasoning: Current Trends*, pages 25–57. Springer, Berlin, Heidelberg.

Levenshtein, V. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics-doklady*, 10(8):707–710.

Levy, O. and Goldberg, Y. (2014). Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL–2014)*, volume 2 (Short papers), pages 302–308, Baltimore, Maryland. Association for Computational Linguistics.

Lo, H.-W., Yifei, Z., Fam, R., and Lepage, Y. (2022). A study of re-generating sentences given similar sentences that cover them on the level of form and meaning. In *Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation (PACLIC-36)*, pages 369–378, Manila, Philippines. De La Salle University.

Luong, T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP–2015)*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Luong, T., Socher, R., and Manning, C. (2013). Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113, Sofia, Bulgaria. Association for Computational Linguistics.

Makarov, P., Ruzsics, T., and Clematide, S. (2017). Align and copy: UZH at SIGMORPHON 2017 Shared Task for morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 49–57, Vancouver. Association for Computational Linguistics.

Mansion, J. E. (1981). *Harrap's New Shorter French and English Dictionary.* George G. Harrap Co. Ltd, London, Paris, Stuttgart.

Marquer, E., Alsaidi, S., Decker, A., Murena, P.-A., and Couceiro, M. (2022). A deep learning approach to solving morphological analogies. In Keane, M. T. and Wiratunga, N., editors, *Case-Based Reasoning Research and Development*, pages 159–174, Cham. Springer International Publishing.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Mikolov, T., Yih, W.-T., and Zweig, G. (2013b). 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)*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

Muller, P., Hathout, N., and Gaume, B. (2006). Synonym extraction using a semantic distance on a dictionary. In *Proceedings of TextGraphs: the First Workshop on Graph Based Methods for Natural Language Processing*, pages 65–72, New York City. Association for Computational Linguistics.

Neubig, G. and Hu, J. (2018). Rapid adaptation of neural machine translation to new languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP–2018)*, pages 875–880, Brussels, Belgium. Association for Computational Linguistics.

Neuvel, S. and Fulop, S. A. (2002). Unsupervised learning of morphology without morphemes. In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, pages 31–40. Association for Computational Linguistics.

Neuvel, S. and Singh, R. (2001). Vive la différence! what morphology is about. *Folia Linguistica*, 35(3-4):313–320.

Nicolai, G., Hauer, B., Motallebi, M., Najafi, S., and Kondrak, G. (2017). If you can't beat them, join them: the University of Alberta system description. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 79–84, Vancouver. Association for Computational Linguistics.

Pedersen, T. (1996). Fishing for exactness. In *Proceedings of the South-Central SAS Users Group Conference (SCSUG-96)*, Austin, TX.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP–2014)*, pages 1532–1543.

Putro, S. C., Jiono, M., Nuraini, N. P., and Fam, R. (2021). Development of statistics teaching materials using augmented reality to reduce misconception. In *2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE)*, pages 151–156.

Rashel, F., Luthfi, A., Dinakaramani, A., and Manurung, R. (2014). Building an Indonesian rule-based part-of-speech tagger. In *Proceedings of 2014 International Conference on Asian Language Processing (IALP–2014)*, pages 70–73, Kuching, Malaysia.

Rashel, F. and Manurung, R. (2013). Poetry generation for Bahasa Indonesia using a constraint satisfaction approach. In *Proceedings of 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS–2013)*, pages 219–224, Bali, Indonesia.

Rashel, F. and Manurung, R. (2014). Pemuisi: a constraint satisfaction-based generator of topical Indonesian poetry. In *Proceedings of the Fifth International Conference on Computational Creativity (ICCC–2014)*, pages 82–90, Ljubljana, Slovenia.

Resnik, P., Olsen, M. B., and Diab, M. (1999). The Bible as a parallel corpus: Annotating the 'book of 2000 tongues'. *Computers and the Humanities*, 33(1):129–153.

Salimans, T. and Kingma, D. P. (2016). Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 901–909. Curran Associates, Inc.

Schone, P. and Jurafsky, D. (2000). Knowledge-free induction of morphology using latent semantic analysis. In *Proceedings of The Fourth Conference on Computational Language Learning (CoNLL-2000) and the Second Learning Language in Logic Workshop (LLL–2000)*, pages 67–72, Lisbon, Portugal.

Schulz, K. U. and Mihov, S. (2002). Fast string correction with levenshtein-automata. *International Journal of Document Analysis and Recognition.*, 5(1):67–85.

Shu, R. and Nakayama, H. (2017). Compressing word embeddings via deep compositional code learning. *CoRR*, abs/1711.01068.

Silfverberg, M., Wiemerslage, A., Liu, L., and Mao, L. J. (2017). Data augmentation for morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 90–99, Vancouver. Association for Computational Linguistics.

Singh, R., editor (2000). *The Yearbook of South Asian Languages and Linguistics-200*. Sage, Thousand Oaks.

Singh, R. and Ford, A. (2000). In praise of Sakatayana: some remarks on whole word morphology. In Singh, R., editor, *The Yearbook of South Asian Languages and Linguistics-200*. Sage, Thousand Oaks.

Soricut, R. and Och, F. (2015). Unsupervised morphology induction using word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT–2015)*, pages 1627–1637, Denver, Colorado. Association for Computational Linguistics.

Stroppa, N. and Yvon, F. (2005). An analogical learner for morphological analysis. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL–2005)*, pages 120–127, Ann Arbor, Michigan. Association for Computational Linguistics.

Wagner, R. A. and Fischer, M. J. (1974). The string-to-string correction problem. *Journal of the Association of Computing Machinery*, 21(1):168–173.

Wang, H., Yang, W., and Lepage, Y. (2014). Sentence generation by analogy: Towards the construction of a quasi-parallel corpus for Chinese-Japanese. In *Proceedings of the 20th Annual Conference of the Japanese Association for Natural Language Processing*, pages 900–903, Hokkaido, Japan. The Association for Natural Language Processing.

Wang, W., Fam, R., Bao, F., Lepage, Y., and Gao, G. (2019). Neural morphological segmentation model for Mongolian. In *2019 International Joint Conference on Neural Networks (IJCNN–2019)*, pages 1–7, Budapest, Hungary.

Wegari, G. M., Melucci, M., and Teferra, S. (2015). Suffix sequences based morphological segmentation for Afaan Oromo. In *Proceedings of the IEEE 12th 2015 AFRICON International Conference*, page (no pagination), Addis Ababa, Ethiopia.

Wintner, S. (2014). *Natural Language Processing of Semitic Languages*, chapter Morphological Processing of Semitic Languages, pages 43–66. Springer, Berlin, Heidelberg.

Yang, C. (2016). *The Price of Linguistic Productivity: How Children Learn to Break the Rules of Language*. The MIT Press.

Yifei, Z., Fam, R., and Lepage, Y. (2022). Extraction of analogies between sentences on the level of syntax using parse trees. In *Proceedings of the workshop Analogies: from Theory to Applications (ATA@ICCBR 2022), held with the 30th International Conference on Case-Based Reasoning*, pages 1 – 13, Nancy, France.

Yvon, F. (2003). Finite-state machines solving analogies on words. Technical report, ENST.

Zhou, C. and Neubig, G. (2017). Morphological inflection generation with multi-space variational encoder-decoders. In *Proceedings of the CoNLL*

SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection, pages 58–65, Vancouver. Association for Computational Linguistics.

Zoph, B., Yuret, D., May, J., and Knight, K. (2016). Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP–2016)*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.