Network Theory applied to Linguistics – New Advances in Language Classification and Typology

by

Olga Abramov

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Dr. rer. nat.
(Computer Science)
in the Bielefeld University
2011

Supervisors:

Professor Dr. Alexander Mehler
Dr. Britta Wrede

ACKNOWLEDGEMENTS

Thanks to the people who made this dissertation possible. Especially I would like to thank my husband Vitali Abramov for his appreciation and support making the preparation of the thesis possible. I thank my whole family Ilya, Ksenia, Maria, mother, father, my dear brother Roman and all my relatives for encouraging me in doing this work. I thank you all for your patience.

My special thank goes to my supervisor Alexander Mehler whose help was indispensable for the preparation of this thesis. Alexander Mehler guided this work and supplied his ideas, solutions and recommendations that were implemented here. Further, I would like to thank my second supervisor Britta Wrede for her support and for agreeing to supervise my work. I would also like to thank Ipke Wachsmuth for his advices and support.

I am grateful to all my friends and colleagues for their advices and help: Kirsten Bergmann, Karina Schneider-Wiejowski, Ulli Waltinger and Rüdiger Gleim, Margret Barner. I thank my proofreader of English Vincent Gouws and my mother Tatiana Lokot who had proved the mathematical formula, and to my father Lev Pustylnikov who had always motivated me to do the PhD. I am very grateful to my deceased friend Leonid Tsypin – his advices were always helpful in difficult times.

TABLE OF CONTENTS

PUBLICATIONS AND CONTRIBUTORS

Many parts of the thesis were pre-published in national and international peer-reviewed journals, conference and workshop proceedings. In this chapter, I list those of my publications that are included partially or in full in this thesis. While the papers are published in collaboration with others, I describe as best as I can the subdivision of work within the papers. The publications are ordered according to the chapter they appear in.

Chapter III

This Chapter is based on the *conference proceedings* publication *Pustylnikov* (2009a). This publication presents a simulation model of derivational morphology which was elaborated by me, and implemented by Roman Pustylnikov as part of his diploma thesis (*Pustylnikov*, 2009b). The two suffix induction algorithms were developed by Roman Pustylnikov in agreement with me. The implementation process was guided by me.

Chapter IV

The analysis of the morphological derivation networks as described in Chapter IV was published as part of the book *Towards an Information Theory of Complex Networks:Statistical Methods and Applications* (*Abramov and Lokot*, 2011) in collaboration with Tatiana Lokot. The paper was written by me. Tatiana Lokot was mainly responsible for Section 4.3.1 which was written in collaboration with me. Both authors are responsible for reediting of the paper.

Chapters VI-VIII

Chapters VI-VIII are based on the *journal publication Abramov and Mehler* (2011). The publication was completed in collaboration with Alexander Mehler who did not only provide the model in form of the *Quantitative Networ Analysis* (QNA) but also the numerical results of the corresponding network indices. Chapters VI-VIII apply this model as well as these numerical results.

All classifications presented in this thesis were performed by MATLAB version 7.2.0.232 (R2006a) using the framework developed in the *AG Texttechnologie* by Rüdiger Gleim, Alexander Mehler, Armin Wegener and Olga Abramov. The treebanks were converted using a unified XML based format that was described in *Pustylnikov and Mehler* (2008); *Pustylnikov et al.* (2008).

LIST OF FIGURES

Figure

LIST OF ABBREVIATIONS

GSDN Global Syntactic Dependency Network

MDN Morphological Derivation Network

PoS parts of speech

RMDN Random Morphological Derivation Network

WS Watts & Strogatz graph

BA Barabasi & Albert graph

ABSTRACT

Network Theory applied to Linguistics – New Advances in Language Classification and Typology
by
Olga Abramov

Supervisors: Prof. Dr. Alexander Mehler, Dr. Britta Wrede

This thesis bridges between two scientific fields – linguistics and computer science – in terms of *Linguistic Networks*. From the linguistic point of view we examine whether languages can be distinguished when looking at network topology of different linguistic networks. We deal with up to 17 languages and ask how far the methods of network theory reveal the peculiarities of single languages. We present and apply network models from different levels of linguistic representation: syntactic, phonological and morphological. The network models presented here allow to integrate various linguistic features at once, which enables a more abstract, *holistic*[1] view at the particular language.

From the point of view of computer science we elaborate the instrumentarium of network theory applying it to a new field. We study the expressiveness of different network features and their ability to characterize language structure. We evaluate the interplay of these features and their goodness in the task of classifying languages genealogically. Among others we compare network features related to: average degree, average geodesic distance, clustering, entropy-based indices, assortativity, centrality, compactness etc. We also propose some new indices that can serve as additional characteristics of networks. The results obtained show that network models succeed in classifying related languages, and allow to study language structure in general. The mathematical analysis of the particular network indices brings new insights into the nature of these indices and their potential when applied to different networks.

Keywords:

Language Classification, Dependency Treebanks, Linguistic Networks, Network Theory, Information Theoretic Measures.

[1]The term will be explained later in the thesis.

CHAPTER I

Introduction

The present thesis presents and discusses new methods in language classification and typology that are based on networks. In Chapter II we give an introduction to the field of research in this area. We study networks on different levels of linguistic representation (morphology, phonology, syntax) and test their potential for linguistic research.

Our starting point is on the level of morphology. In Chapter III we address the question of morphological productivity of languages by simulating the emergence of derivational rules during language acquisition. We assume that modeling the human learning behavior can help to enhance methods in language processing and retrieval. Children acquiring the first language observe the adults' speech before learning how to express themselves. Learning is a gradual process of acquiring single sounds (phonology), words (lexis), and more complex constructions (morphology, syntax) (*Tomasello*, 2005). Newly learned material is acquired and recognized by already existing knowledge; similarities on each linguistic level contribute to the recognition of new words (*Bybee*, 1988). Despite these observations common simulation models do not consider morphological and phonological information within automatic learning processes. Here, we extend the scope of these models focusing on derivational morphology as a means of language comprehension and production. We present an evolutionary game-theoretic (EGT) (*Lewis*, 1969) framework that explores the emergence of derivational morphology in a multi-agent computer simulation. The system can be used to study morphological productivity of languages. Furthermore, mechanisms driving the acquisition of morphological competence can be examined by means of this simulation. In the following chapter we use the simulation model to induce morphological derivation networks of different languages.

Chapter IV presents a network-theoretic approach to morphology. In particular, we introduce a network model of derivational morphology in languages. We focus on suffixation as a mechanism to derive new words from existing ones. We induce networks of natural language data consisting of words, derivation suffixes and parts of speech (PoS) as well as the relations between them (using the morphological derivation game presented in Chapter III). In measuring the entropy of these networks by means of so called information functionals we aim to capture the variation between typologically different languages. Thus, we rely on the work of *Dehmer* (2008) who

has introduced a framework for measuring the entropy of graphs. In addition, we compare several entropy measurements recently presented for graphs. We check whether these measurements allow us to distinguish between language networks, on the one hand, and random networks, on the other. We found out that linguistic variation among languages can be captured by investigating the topology of the underlying networks. Furthermore, information functionals based on distributions of topological properties turned out to be better discriminators than those based on properties of single vertices.

In Chapter V, we deal with phonological networks. We consider the phoneme inventories of languages and approach to reconstruct genetic relationships by means of them. The idea behind this approach is that the relationship between two languages, after they have split apart, continues to exist in their sound systems. But this is not necessarily the case. The processes of phoneme inventories' change can follow different rules so that two genetically related languages can be completely dissimilar in terms of phonology. In Chapter V we ask whether phonological similarities, computed based on phonological networks, also match the genetic relationships of languages. Moreover, we examine whether phonological closeness allows us to distinguish one language family from another, and which families are more or less distinct from each other. In addition, we look at the inner-family similarities between languages with respect to their phonology. Are languages within a family similar to each other phonologically or not, and if not, what could be the reasons for this? Finally, we study phonological similarities between pairs of particular languages (across language families) and compare the results with typological findings for these languages obtained in other studies. All the approaches presented and discussed in this chapter could help to shed light on the change of phoneme inventories, on the one hand, and the genetic distance of languages, on the other.

Chapters VI-VIII present an approach to automatic language classification by means of syntactic networks. Networks of up to 17 languages were constructed from dependency treebanks (Chapter VI), and the topology of these networks (Chapter VII) serves as input to the classification algorithm (Chapter VIII). The results match the genealogical similarities of these languages. In addition, we test two alternative approaches to automatic language classification - one based on n-grams and the other on quantitative typological indices. The results of the three methods are compared. The network-based approach, though rather complex to compute, produces the best outcomes. Beyond genetic similarities, network features (and combinations of them) offer a new source of typological information about languages. Thus, network features can be studied in combination with others as well as in isolation providing an additional means for typological research.

In Chapter IX, we summarize the thesis and give final conclusions.

CHAPTER II

Language Classification and Typology

2.1 Introduction

In general, studies in language classification try to uncover similarities between languages that, according to *Daumé III* (2009), can be related to one of the following areas:

- genealogical relations
- linguistic universals
- areal effects
- chance

Genealogical relations result from the assumption that languages have split apart in time descending from a single ancestor language. In this case, they can be grouped together to language families in which the common properties are still observable. Some common properties in languages, however, can be explained by linguistics universals (*Greenberg*, 1966). These are properties that are more common in some languages than in others. There are also properties occurring just by chance. And finally, there are properties that are shared among not necessarily genetically related languages simply because they are situated close to each other. These are called *areal* effects.

In this chapter we report issues regarding the first three reasons of language relationship listed above - *genealogical*, *typological* and *areal* effects. This is done, in order to provide a base for interpretation of the results of the network classifications referred to in this thesis. Furthermore, we discuss recent studies dealing with genealogical, typological, areal and in particular *network-based* classifications.

2.1.1 Genealogical Language Reconstruction

One research direction dealing with language classifications is the area of language reconstruction. This research field attracted researchers from various disciplines: physicians, biologists and linguists (especially historical-comparative linguists). The leading assumption here is that all languages originate from a single proto-language which has been split apart into smaller pieces or language families.

> "If two or more languages share a feature which is unlikely to have occurred spontaneously in each of them, this feature must have arisen once only, when these languages were one and the same."(*Anttila*, 1972)

Different methods were proposed to recover genetic relationships of languages, which are mostly based on *lexicostatistics* (e.g. *Swadesh* (1952); *Batagelj et al.* (1992); *Bryant et al.* (2005)). This means that the number of common basic words (cognates[1]) determines the degree of distance between languages. The more words two languages share, the closer their genetic relationship is assumed to be. Genetic relationships are represented in terms of trees leading to the proto-language. In a nutshell, the lexicostatistical approach determines the proportion of the most basic vocabulary shared by two languages (*Warnow et al.*, 1996). The validity of this method was widely questioned since it disregards many factors in language. Alternative approaches include additional information like phonology, morphology, etc. to calculate genetic trees.

Oswalt (1970) was one of the first to provide an approach to automatically classifying languages. He classified seven Indo-European languages comparing lists of cognates according to phonological characteristics of their realization. His goal was to rule out the chance resemblances between cognates, and to estimate the expected number of common phonological characteristics two languages that are related should have. *Oswalt* (1970)'s method allows to detect "interrelationships of the accepted members of the Indo-European stock" (*Oswalt*, 1970, 125). Additionally, the results suggest an association between the Indo-European and Finnish (Uralic) languages. However, since the method is based on word-lists, lexical change (e.g., borrowing from other languages) could have biased the result. Thus, additional investigations about the relatedness between these language families, especially concerning other levels of linguistic comparison, are needed.

Warnow et al. (1996), proposed a combined approach to language reconstruction using cognates, morphological and phonological features to reconstruct the tree.

Batagelj et al. (1992) enhanced the cognate-based method providing simple distance metrics to measure the similarity between cognates. For example, they calculated the number of different characters needed to transform one form of a cognate from one language into another form of the same cognate from the other language. They clustered 65 languages based on these counts and achieved good classification results comparable to results achieved applying the historical explorative reconstruction methods. A more elaborated approach using normalized edit distances and graph walks was proposed by *Blanchard et al.* (2009) who extended the sample of Swadesh including Austronesian languages.

Holman et al. (2008) presented an approach to automatically classifying languages based on word-lists, which are not restricted to cognates. They developed several techniques to identify the most stable words that improve the classification. This

[1]Cognates are pairs of words from different languages that originate from the same ancestor language. The common origin is determined by regular phonetic change from one language to another and by related meaning of the two words. Borrowed words are not cognates (*Kruskal et al.*, 1992).

way, they reduced the word space from 100 to 40 word features. They also reported that combining word-list based features with typological features from the **W**orld **A**tlas of **L**anguage **S**tructures (WALS) (*Haspelmath et al.*, 2005) can improve the outcome of the classification.

Sullivan and McMahon (2010) followed the principle of Swadesh applying *phonostatistics* instead of lexicostatistics in order to compare Germanic dialects. The authors quantitatively analyzed "the phonetic realisations of cognates in different languages/varieties."[2] Various methods to compute the phonetic distance between pairs of cognates were computed. The distances were used to construct the phylogenetic trees of Germanic languages and varieties.

Other algorithms to automatically reconstruct language relationships, which are based to a large extent on phonetics, are reviewed in *Kondrak* (2002).

All the approaches discussed above restrict the feature space to lexical or phonetic correspondence to establish relationships among languages (cf. *Holman et al.* (2008) for typological and *Bakker et al.* (2009) for combined lexicostatistical and typological approaches). However, as we know from areal linguistics as well as from studies on language change it is not sufficient to explain the structure of a particular language only by means of its genetic origin. Thus, relatedness between languages must consist of more than just the percentage of common words, phonemes or N-grams.

Daumé III (2009) has shown that methods in language reconstruction can be enhanced by including areal information. *Daumé III* (2009) identified those typological features from WALS which are shared among *arealy* related languages and used this information to improve the reconstruction of genetic trees.

The above approaches deal with selected features from several levels of linguistic representation. However, a *general classification* of languages according to *Altmann and Lehfeldt* (1973) is one that captures as many levels as possible. Since it is hardly feasible to integrate all features from all linguistic levels, another way is to extract an abstract model from language data allowing to examine the different linguistic properties by means of a general imprint of a language. In the present dissertation we present various approaches to data driven language classification by means of networks. We discuss how network characteristics can be used to identify relationships between languages reflecting typological, genealogical or areal differences.

2.1.2 Language Recognition

Language classification is somehow related to the field of *Language Recognition* (LR). LR applies several techniques to guess the language of an input text (or speech) in order to solve an information retrieval task. Methods in LR use common words (*Grefenstette*, 1995), closed word classes (*Lins and Gonçalves*, 2004), single characters (*Churcher et al.*, 1994; *Takci and Sogukpinar*, 2004) or N-grams (*Cavnar and Trenkle*, 1994; *Combrinck and Botha*, 1995; *Dunning*, 1994; *Ahmed et al.*, 2004; *McNamee*, 2005) as features to distinguish languages. Although these approaches perform well in recognizing languages, the features introduced reflect frequencies of word forms or

[2] *Sullivan and McMahon* (2010, 327).

character sequences rather than distinct typological properties of a language.

The crucial point concerning these approaches is the fact that they are all supervised. That means that, first, the classifier is trained on some data, and then new data pieces are categorized in comparison to the learned classes. This way, languages can be recognized with no suggestions about typological similarities of these languages having been made. To speak about similarities in the context of LR means to understand similarity as the amount of common features (e.g., words or N-grams) shared by these languages. Even so, restricting a language classification task to the level of word or character frequencies is not sufficient from the typological point of view since two languages can be alike according to their character sets but may differ, for example, in their grammatical behavior.

We implement the N-gram based approach (*Cavnar and Trenkle*, 1994) (NG) and apply it to our data in order to compare the outcomes to other (network-based and typological) approaches. We check the genealogical as well as the typological performance of this method using a data set of 11 languages. The NG approach will be described later in Section 8.2.1.

2.2 Directions in Typology

Another direction of research exploring similarities among languages is the area of language typology. Current typological research can be roughly divided into *holistic* vs. *partial* approaches. The holistic thinking in typology, which comes close to the idea of a *general language classification* of *Altmann and Lehfeldt* (1973), appeared earlier in history and was influenced by genealogical research, on the one hand, and by developments in natural science, on the other. The main assumption in this phase was to view language as a whole (*holistic*) organic unit (biology) or system (physics, system theory, synergetic linguistics) consisting of interrelated components (*Whaley*, 1997). The first typological research put forward by 19th century linguists Wilhelm von Humbold (1767-1835), Friedrich von Schlegel (1772-1829) and August Schleicher (1821-1868), amongst others, was highly morphology-oriented (*Whaley*, 1997). August Schleicher pointed out that analytic (or isolating) languages with little of morphology exhibit a more restricted word order than synthetic languages (*Masayoshi and Bynon*, 1995). Thus, first attempts at language classification aimed to assign a language to a particular morphological *type* (e.g. analytic vs. synthetic, agglutinating vs. fusional). These classifications turned out to be insufficient since pure types do not generally occur - languages are mostly mixed, making clear divisions into types difficult.

The Prague School linguists (*Skalička*, 1979) as well as Edward Sapir shifted from this classical perspective to a gradual one, assigning a language, for example, a degree of fusion, analysis, etc.. There are up to five possible types (or extremes) distinguished by (*Skalička*, 1979): synthetic (inflectional), isolating (analytic), polysynthetic, agglutinating and introflexive. These types are defined by sets of co-occuring typological properties, for example, the analytic type can be distinguished by an absence of affixation, by a fixed word order, etc.. This means that languages are assigned to

types by means of combinations of typological features, in contrast to the classical dichotomies, for example, a*nalytic* vs. *synthetic*. For instance, Czech and Russian are defined as *highly synthetic* by Skaliča although both languages differ in the occurrence of *analytic* and *agglutinating* properties. In sum, since it is impossibile to make clear distinctions, typologists turn to quantitative approaches measuring the relative similarity among languages.

The main goal of holistic typology is to characterize language in general in order to enable predictions about the language type based on some particular characteristics (*von der Gabelentz*, 1901). Because of the complexity of this task, typologists started to concentrate on sub-parts rather than on language as a whole. *Partial* approaches aim to compare languages based on single (not only morphological) grammatical phenomena like case-marking, word order, relative clauses, passives, etc. (*Masayoshi and Bynon*, 1995). Findings in this area greatly enlarged our knowledge about particular languages and language structures in general. One problem, as clarified by *Hawkins* (1983), is that researchers tended to lose sight of the goal of integrating these findings into the whole language system over time:

> "[...] small pieces of language are plucked out from the overall grammar that contains them, and the range of attested variation is described, and universal generalizations, or truths, are proposed that are compatible with all and only the observable patterns. Obviously, the more such pieces of language we study, the more universal generalizations we gain. But it is not clear that we are making much progress towards understanding how the variants that an individual language selects in one area of grammar are determined by, or determine, the variants that it selects in another."

Although the work of Greenberg, who found many correlations among linguistic characteristics, supports the understanding of language as a system, a large amount of present-day research in typology is concerned with investigations of single phenomena only. Observed phenomena differ in their 'predictive power' with respect to the whole structure of a language, that is, some phenomena can be subordinated to, or are natural consequences of other phenomena. For example, dominant word order types can be explained by means of the *degree of analysis*. Thus, language can be understood as an "interrelated network allowing for hierarchical structuring" (*Masayoshi and Bynon*, 1995). It is desirable to find an evaluation base for typological studies allowing researchers to judge the predictive power as well the interrelatedness of single typological phenomena (*Sgall*, 1995).

2.3 Language Change and *Areal* Effects

The importance of areal aspects and effects of borrowing from one language are emphasized by researchers in language classification (*Warnow et al.*, 1996; *Daumé III*, 2009). Misclassifications on the genealogical level are often attributed to areal effects (*Daumé III*, 2009).

The areal approach can be best described in contrast to the other two approaches: genealogical and typological. Whereas the genealogical approach explains similarity by means of genetic relations, the areal perspective focuses on geographic closeness and tries to explain similarity, even among genetically not related languages, by means of language contacts. The methods to establish similarity are the same as used for genealogical or typological classifications. *Klimov* (1980) describes the three perspectives in the following way: the genealogical one reflects genetic relationships, the typological - the diversity of types and the areal - historical convergence of languages.

Studies on areal effects can be traced back to the work of Trubetzkoy in the 1920s. These studies examine similarities between languages within particular areas, like for example, the *Balkans*. Questions regarding areal linguistics are: How is a linguistic area constituted? How many languages does it comprise? Areal language groups are more fuzzy and less specified than the genealogical ones (*Thomason and Kaufman*, 1988). How many features or "traits" are needed to describe an area? How closely must the languages be situated to "interact"? Are some features more easily borrowed than others? (*Campbell*, 2006).

(*Daumé III*, 2009), for example, tested two statistical models - Pitman-Yor process (flat) and Kingman's coalescent (hierarchical) to determine linguistic areas and the areal features among 129 linguistic features from WALS and 2150 languages. Further, he reconstructed genetic trees enhancing the results of *Teh et al.* (2009) by including areal features. His results are compliant with the hierarchy of borrowability of linguistic features across languages. That is, features from WALS identified as areal features are those that are most easily borrowed according to the hierarchy of *Ross* (1988) (nouns > verbs > adjectives > syntax > non-bound function words > bound morphemes > phonemes).

2.4 Related Work on Language Networks

As mentioned in the previous sections networks appear as appropriate models of language since they allow to account for the complexity of linguistic relations.

Kello and Beltz (2009) presented a network model of language based on its orthographic (or phonological) lexicon. The main objection behind this study was to examine the *principle of least effort* (*Zipf*, 1949) on the level of lexical material. This principle postulates a communicative trade-off between the speakers' and listeners' needs to minimize the memory effort (i.e., storing new words) and the disambiguation effort (i.e., distinguishing the actual meaning of the word). Basically, two antipodal forces go together: the force to minimize the memory capacity (the extreme case is one word for all meanings) and the force to be maximally distinctive (the extreme case is one word per meaning). Since both forces cannot reach their maximum when acting together, languages tend to find an optimal solution to satisfy both to the greatest possible extent. On the level of the distribution of word forms, this behavior was shown to follow a power law as observed by *Ferrer i Cancho and Solé* (2003). *Ferrer i Cancho and Solé* (2003) could show that varying the parameters a) the number of words, and b) the number of meanings per word, the power law behavior of

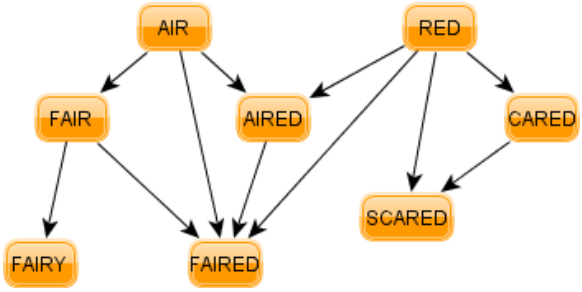

Figure 2.1: An excerpt from the lexical wordform network taken from *Kello and Beltz* (2009).

the system appears only close to the transition between the two antipodal forces. Transferred to language, where we observe the same scaling law of word frequencies we can see this transition.

Kello and Beltz (2009) tested the principle of least effort on the distinctiveness of lexica of various languages. Their primary observation is that lexical items normally do not represent atomic units, but consist of the lexical material already present in language (words, suffixes, etc.). Here the disambiguation effort would force a maximal distinctiveness of words and, thus, disallowing, for example, the creation of compounds. The example of an unambiguous word in English, according to *Kello and Beltz* (2009), is the word YACHT, since neither YACHT nor ACHT nor YAC are part of the English lexicon. The word FAIRED, on the other hand, is less distinctive while it contains FAIR, AIR, AIRED, IRE and RED. The authors constructed a directed network of atomic words (like RED) as the roots and composite words, such as 'leaves' in order to test the power-law behavior of out-degrees. An excerpt of this network is shown in Figure 2.1. Of course, this model does not consider grammatical morphemes, which can also be highly recurrent in language, and can function as disambiguation marks (e.g., for different cases of the word). According to *Bybee* (1988) language is represented as a network of phonetic, phonological, morphological, etc. elements, and not just arbitrary symbols. Thus, in order to be precise one should consider these elements too. However, if together with *Köhler* (1986) we assume a synergy between different linguistic levels, single levels (such as lexical, morphological, etc.) can be examined in isolation and compared to each other in order to verify this synergy. Network models gained a special interest in recent studies on language typology and classifications (cf. *Choudhury and Mukherjee* (2009) for a review of network models on different linguistic levels, i.e., phonology, syntax, etc.) due to their potential in modeling language complexity and dynamics.

Ferrer i Cancho et al. (2004) were the first to study the properties of syntactic networks based on data from dependency treebanks of three languages. They could show that the topology of these networks is not random. Rather these networks all match the small world model (SWM) of *Watts and Strogatz* (1998). The work of *Ferrer i Cancho et al.* (2004) helped to shed light on the relation between the degree distributions of syntactic networks and the Zipfian distribution of word frequencies.

Ferrer i Cancho et al. (2007) confirmed the correspondence to the SWM for six languages.

Liu (2008) looked at the topology of language networks of a single language consisting of two different text types. He confirms the congruence with to the small-world model for both text types (networks) which he relates to the Zipfian law of natural language. He also observed small differences in the values of the coefficients for treebanks representing different text types.

Liu et al. (2010) studied the question whether local differences in syntactic annotation scheme (different representation of coordinating constructions) influence the global structure of dependency networks. They found out that global properties of small-worldness and scale-freeness are not significantly influenced by local syntactic changes. However, other network properties like centrality are more sensitive to local changes of particular syntactic constructions. *Liu et al.* (2010) argued that we need to find other global statistical properties which more clearly local changes in the network. The present thesis examines 21 different network indices with respect to their potential in distinguishing language networks.

Minkov and Cohen (2008) performed a graph walk based on named entity extraction (or *named entity coordinate term extraction*) using directed weighted labeled Global Syntactic Dependency Networks (Global Syntactic Dependency Network (GSDN)).[3] They have shown that sequences of labeled dependency paths bear information about word similarities allowing to detect location and person names.

Mehler (2008a) introduced *Quantitative Network Analysis* (QNA) as an approach to classify complex networks in terms of their topology. QNA combines complex network theory with unsupervised machine learning to model classifiers of networks that explore only their structure. This is exemplified by classifying social and linguistic networks - the latter derived from the textual and lexical level - where all these networks are derived from special Wikis. The classification shows not only that these networks can be distinguished ontologically, but also functionally in terms of communication areas.

Mehler et al. (2010a) further applied QNA to classify languages genealogically. As their data source *Mehler et al.* (2010a) utilized the category system of Wikipedia that is available in many languages around the world. They have shown that languages can be classified into language families by exploring the topology of the Wikipedia category systems of the corresponding languages to be classified.

In this dissertation, among others, we apply QNA to a data driven language classification by means of syntactic networks (Chapters VI-VIII). We aim to find out whether the network structure induced from dependency treebanks provides any information about the relatedness of languages analogous to the classification presented in *Mehler et al.* (2010a). While *Mehler et al.* (2010a) used social ontologies and, thus, semantic networks to classify languages, we will use syntactic dependency networks for the same task. *Liu and Xu* (2011, 28005-1) argue that

> "The complex-network approaches can obtain language classifications as

[3]This notion goes back to *Ferrer i Cancho et al.* (2004) and will be explained in more detail below.

precise as achieved by contemporary word order typology."

Liu and Xu (2011) present an approach to automatic language classification and typology very similar to QNA but use a smaller amount of network indices and fewer networks. The results obtained by *Liu and Xu* (2011) are very valuable for us, since they allow to (partially) compare our outcomes to those obtained in their study. We will come back to the study of *Liu and Xu* (2011) in Chapter VIII when discussing the syntactic network-based classifications.

2.5 Summary

In this thesis we aim at a *holistic* typology allowing to make statements about a language as a whole based on different linguistic levels represented as a network. We follow *Liu and Xu* (2011, 28005-1), who suggest that

> "[...] linguists have to resort to new methods to study languages from a network perspective, which focusses on the overall picture of a language rather than the structural details."

Studies discussed in this section suggest that the network perspective can enhance the understanding of complex linguistic relations or the discovery of relations not observed so far. Areas such as language classification, identification, reconstruction, typology, etc. can benefit from the application of network models to language.

The present thesis analyzes various network models applicable on the levels of morphology (Chapter IV), phonology (Chapter V) and syntax (Chapter VI ff.).

CHAPTER III

Modeling Learning of Derivation Morphology in a Multi-Agent Simulation

3.1 Introduction

How do children acquire language? The most striking fact in language learning is the ability of humans to infer grammatical relations that are not explicitly learned. Sometimes this fact is called "the poverty of the stimulus" that children easily overcome in comparison with non-humans (animal or artificial subjects). There are various explanations for this ability in the literature. *Chomsky* (1965) explains this ability of human beings by an innate universal grammar which is instantiated with specific parameters of a particular language. By contrast, the functionalist perspective assumes a dynamic grammar construction based on language use (*Tomasello*, 2005). In this second approach learning has a greater importance for language acquisition than innateness (*Ford and Voegtlin*, 2003). We argue for the latter approach and claim, together with *Bybee* (1988), that regularities are not explicitly learned but rather automatically induced from the input language of the communication partner. Newly learned material is acquired and recognized based on already existing knowledge while similarities on each linguistic level contribute to the recognition of new words (*Bybee*, 1988). This process can be described as a network with linguistic items (words, morphemes, etc.) as vertices and usage-relations as edges. In the next chapter we present a morphological derivation network model addressing this issue.

In this chapter we provide a base for induction of morphological networks by means of a multi-agent simulation. We concentrate on the acquisition of derivation rules for the main *parts of speech* (POS)[1] as the dependent variable and the input language as the independent variable in a child-adult simulation game. The input languages are varied - we test the model on two natural languages (English and German) and on a randomly generated word set. In principal, any language can be fed into the system. If a language does not use suffixation to derive new words, more training will be needed to learn the correct word-to-POS mappings (since no regularities within the word can be detected).

[1]The terms "POS" and "word class" are used interchangeably throughout the thesis.

The theoretical background of the presented system is described in Section 3.2. The general scenario of the simulation is explained in Section 3.3. The experimental procedure is described in Section 3.4. Results are presented and discussed in Section 4.5. Finally, we give some conclusions in Section 4.7.

3.2 Theoretical Background

In the literature, two mechanisms of morphological processing are outlined (i.e., the *dual route model*): the full word route and the parsing route. Both mechanisms are supposed to depend on word frequency. A highly frequent word is more accessible to the mental lexicon and can be easily recognized as a whole (full word route) without decomposition into stem and affixes. Processing a less frequent word, decomposition, in turn, might bring additional support and facilitate the recognition task (parsing route). A suffix, for example, that is often used to derive one word (class) from another can facilitate the processing (e.g., *ease* > *eas-y*, i.e., adjective from verb). In general, each piece of information that can be identified within and beyond the word is utilized in processing.

In languages that make use of derivational morphology, derivational suffixes are used to derive, for example, an adjective from a noun. In some languages (like in German) there are more than one suffixes forming the same word class, which are supposed to compete during the evolution of language. Productive suffixes are those that are used more frequently in word formation than other (*Baayen*, 1992). However, this assumption is controversial since some suffixes reflect different semantic functions in language. They are restricted to different semantic domains of the same word class, and thus, they are not in competition. For example, two derivations *'Gleich-ung'* and *'Gleich-nis'* of the stem *'gleich-'* produce different semantic meanings in German. Furthermore, some derivation suffixes can be fossilized; they are no longer used for production, although, many words formed by these suffixes exist.

In this thesis, we neither make semantic restrictions to the input language nor claim for a real competition of suffixes. However, the presented model allows to test these assumptions for a particular language by examining the identified suffixes.[2]

In an *Evolutionary Game Theoretic* framework (*Smith*, 1982) child-agents C speak to an adult-agent A and to each other. There are word forms F and meanings M, the meaning space is restricted to the main word classes. At the beginning, the adult is selected as a sender S, and it utter a word to a randomly selected receiver-child R. The receiver tries to guess the word class of the uttered word and when successful, becomes the sender. The game is inspired by the research in first language acquisition in which children are faced with the problem of segmenting adults' speech and inducing regularities in order to facilitate the interpretation. The segmentation emphasized in this game concerns the word level and utilizes the regularities within the word to facilitate guessing the word class. In evolutionary game theoretic terms, the population is the number of possible suffixes frequently occurring in words of a

[2]Cf. *Pustylnikov and Schneider-Wiejowski* (2009) who study morphological productivity of German suffixes using this simulation model.

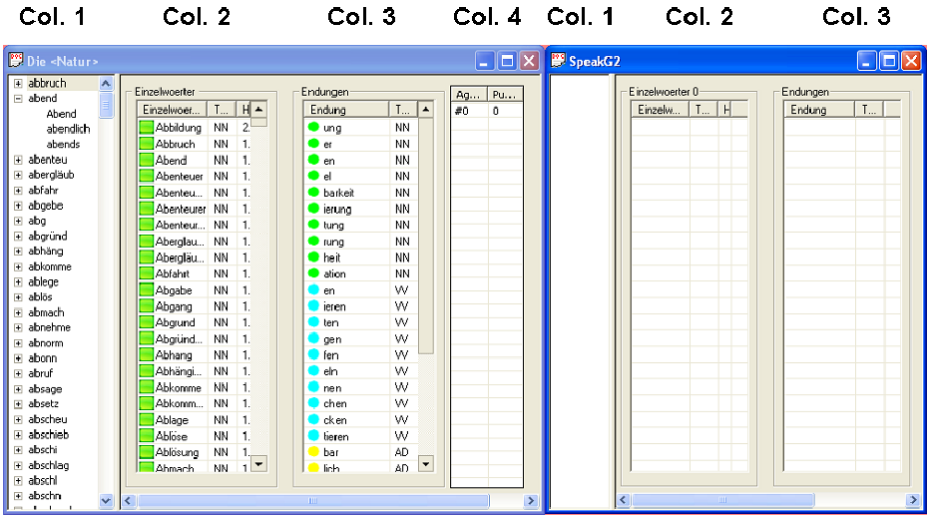

Figure 3.1: The adult (A) dialog box (left) and the empty child (C) dialog box (right) (*Pustylnikov*, 2010).

concrete word class. Successful suffixes that improve the recognition are *replicated*, that is, reused in future conversations. We survey which suffixes actually "win the race" in the natural as well as in the random scenarios.

The difference of this model and the common evolutionary game theoretic models lies in the ability of the agents to induce the word class of the word via decomposition.[3] Here, agents segment words into derivation suffixes and stems. This mechanism is complementary to the full word recognition in language processing according to the *dual route model* (see the previous section).

3.3 Game Setting

In this section we describe the implementation of the game. The agent architecture is described in Section 3.3.1. The decomposition algorithm the agents use is presented in Section 3.3.2. Finally, Section 3.3.3 summarizes the overall game scenario.

3.3.1 Agent Architecture

The snapshot of the system in Figure 3.1 illustrates the different types of agents at the beginning of the game.

3.3.1.1 Adult agents

The adult window consists of four sections. The first column containing a tree view shows the **derivation network**. It lists all possible word derivations of a single word with the corresponding stem. For example, the sub-tree of the word "Abend"

[3] Cf., e.g., *Vogt* (2005); *Gong et al.* (2009) for simulation models using compositionality.

(evening) is *abend* → *Abend* → *abendlich* → *abends*. The derivation network is a subset of the **lexicon** containing all single words of the adult with the corresponding word classes and frequencies that are listed in the second column. The third column lists the suffixes according to their priority (see Section 3.3.2 for a description on how these priorities are obtained) and their word class. The last column maintains the statistics about the communication scores of the agents.

In order to load the derivation network and the lexicon, two kinds of data are required. The derivation network is constructed from a text file in which each line corresponds to a single "word family"[4], for example:

```
Abhang 0 0 abhängen 1 0 abhängig 2 0 Abhängigkeit 0 0
```

A word is followed by two numbers, the first representing the word class (0 = noun, 1 = verb, 2 = adjective) and the second its frequency. Frequencies are attributed to the use of this word during the game, thus, they are initially set to 0 in the adults' derivation network.

We have created the German derivation network manually using `www.canoo.net` and the English network by consulting a dictionary. At first glance, it seems rather awkward to create such a word-family file each time for a language. The initial German file contains 421 entries, accordingly, 421 word families. However, since we have a much smaller amount of word families for English (only 49 entries), we selected a random subset of the German derivation network containing 49 entries (see the results for all the three networks in Section 4.5).

The second text file is employed to construct the lexicon. This file contains a previously tagged text corpus (tagged with word class information, e.g., by TreeTagger[5]). This output file is launched when creating the adult, and all the words that are nouns, verbs or adjectives (+ adverbs) are stored in the lexicon together with their frequencies of occurrence. These frequencies are later used to compute the probability with which a word is uttered by the agents.

In this version of the system, we restrict the word classes to three main classes: verbs, nouns and adjectives. Adverbs are subsumed to the class of adjectives constituting a more general class of attributes. This simplification is made since derivation processes are mostly observable for these main word classes. However, the model can be easily extended to other word classes or integrated as a module into a more sophisticated decomposition system.

3.3.1.2 Child agents

In contrast to the adult, the child agents have a three column window (Fig. 3.1, on the right) . All fields are empty at the beginning. In analogy to the adult agent, the child's first sub column is reserved for the derivation network and the second for the lexicon. The third column will contain the suffixes induced by analyzing single words. All three parts are filled during the game.

[4] A *word family* is a group of words which share a common stem and have a common origin. Words within a family can be spread over different word classes (*Römer and Matzke*, 2005).

[5] `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html`.

3.3.2 Decomposition Algorithm

Each agent (adult or child) is endowed with an internal decomposition mechanism that contains no built in or learned information about a particular language. Thus, in principal any language can be tested by the system. The algorithm utilizes two sorts of information: the lexicon of single words with the corresponding word classes and the derivation network. For the adult, these sources are loaded from the fileswhile children construct them on the fly by inducing the regularities from words uttered by the adult. Note that the availability of the derivation network which might be hardly obtainable for a particular language is not required (cf. the small size of the English and the reduced German networks). In this section we describe two basic algorithms[6] used here to obtain the suffixes from the derivation network and from single words.[7]

3.3.2.1 Suffix Induction from the Derivation Network

A derivation network of an agent consists of sub trees representing a common stem (if available) and the possible derivations. For the word "anerkennen" (acknowledge, appreciate), for instance, the tree would consist of the stem "anerkenn-" and the derivations "anerkennen" (*accept* as a verb) and "Anerkennung" (*acknowledgement, appreciation* as a noun). The algorithm does the following:

i. For each derivation tree D (e.g., *anerkennen*, *Anerkennung*) a common stem $St = l_1 \cdots l_n$, where $l_1, ..., l_n$ are letters, is identified by comparing the words letter by letter. Thus $St \bigcup D$ is the common part of all words in D (that is *anerkenn-* in the above example).[8]

ii. Capitals are converted to lowercase letters and umlauts are converted to ASCII. When the stem is identified, the resulting suffixes are added to the suffix collection of the corresponding word class.

iii. Finally, the best (i.e., most frequent) suffixes are merged with the best from the lexicon.

3.3.2.2 Suffix Induction from Lexicon

To find the significant suffixes from the collected set of words four filtering steps are performed. At the beginning all the words are sorted by word class $c \in C$ and each group is analyzed separately. Formally, a lexicon $\mathbb{L} = \bigcup_c L(c)$ with $L(c)$ as a set

[6]Both algorithms were developed by Roman Pustylnikov as part of his Diploma thesis (*Pustylnikov*, 2010).

[7]The algorithms presented in the following sections use similar techniques as in *Harris* (1967); *Goldsmith* (2001). We do not compare the proposed method to these approaches since we are primarily interested in derivation morphology here. However, future work should evaluate the performance of the decomposition mechanism introduced in this thesis contrasting it with other methods in this area.

[8]Of course, the morpheme analysis of 'anerkennen' presented here is not complete since pre-, in- and circum-fixes are not considered. The example is chosen to illustrate the analysis of suffixes.

of words of class c is given. That is, a lexicon $\mathbb{L} = \{\langle w, c\rangle\}|w \in W, c \in C\}$ consists of pairs $\langle w, c\rangle$ of words and classes. Each word $w = l_1 l_2 \cdots l_n$ is represented as a letter string (*Ukkonen*, 1995). A suffix s is a substring $s = l_j \cdots l_n$ of w, with $j \geq n - 7$.[9]

For each word class c every word from the lexicon is analyzed and the relevant suffixes are filtered out by the following algorithm.

i. For each word all suffixes of the maximal length of 8 are extracted (e.g., for the word *apple:* $e > le > ple, ...$). The occurrences of all suffixes in all other words from the lexicon are counted. Suffixes with a frequency of more than 20% are stored in the $group\_list_w$.

ii. If two suffixes in the resulting $group\_list_w$ have similar frequencies (with respect to a similarity threshold), and one of them contains the other, the shorter is removed from the list (e.g., if *'isch'* and *'ch'* have the same frequency, then *'ch'* is removed).[10]

iii. The $group\_list_w$ is added to the $list\_from\_all\_words$ which contains all suffixes from the other words, and the procedure of step ii. is repeated using a more sensitive similarity threshold.

iv. For each suffix s a list is constructed that contains all suffixes that include s (e.g., the list for $s =$ *'ch'* might contain *'ich', 'ach', etc.*) and which contains no suffixes that have a common part larger than s (e.g. *'klich'* and *'rlich'* are removed from the list of *'ch'*, since they are already in the list of *'lich'*). All suffixes are ranked according to the number of different suffixes in their lists.

The highest number of different suffixes in this list gives the best representative suffix for the word class c. The most productive suffixes are those that are included in several different combinations. The filtering steps ensure that redundant suffixes are removed so that they do not influence the final result.[11]

3.3.3 Game Procedure

The overall game scenario (Fig. 3.3) represents communicative acts among adult-child and child-child. The adult speaks in turn to each of the children. They randomly select a word from their derivation network (1. stage) or from his lexicon (2. stage) and ask a child to guess the word class of this word. Optionally the frequency of the word is considered, that is, words with a higher frequency are more likely to be selected (see the option "`Haeufigkeitsberuecksichtigung bei der Worteinspeisung`" in Fig. 3.2). Varying this parameter allows for testing the role of word frequency in correlation with the emergence of productive derivation rules according to the *dual route model* (see, e.g., *Baayen* (1992)).

A single game run comprises a predefined number of iterations (i.e., the option "`MaxCounter`" in Fig. 3.2). The iterations slightly vary according to the current

[9] The maximal suffix length of 8 was found heuristically, and can be changed on demand.

[10] This is done in order to obtain the longest match, i.e., the actual suffix and not only its parts.

[11] See Figure A.1 in the Appendix A for the complete algorithm.

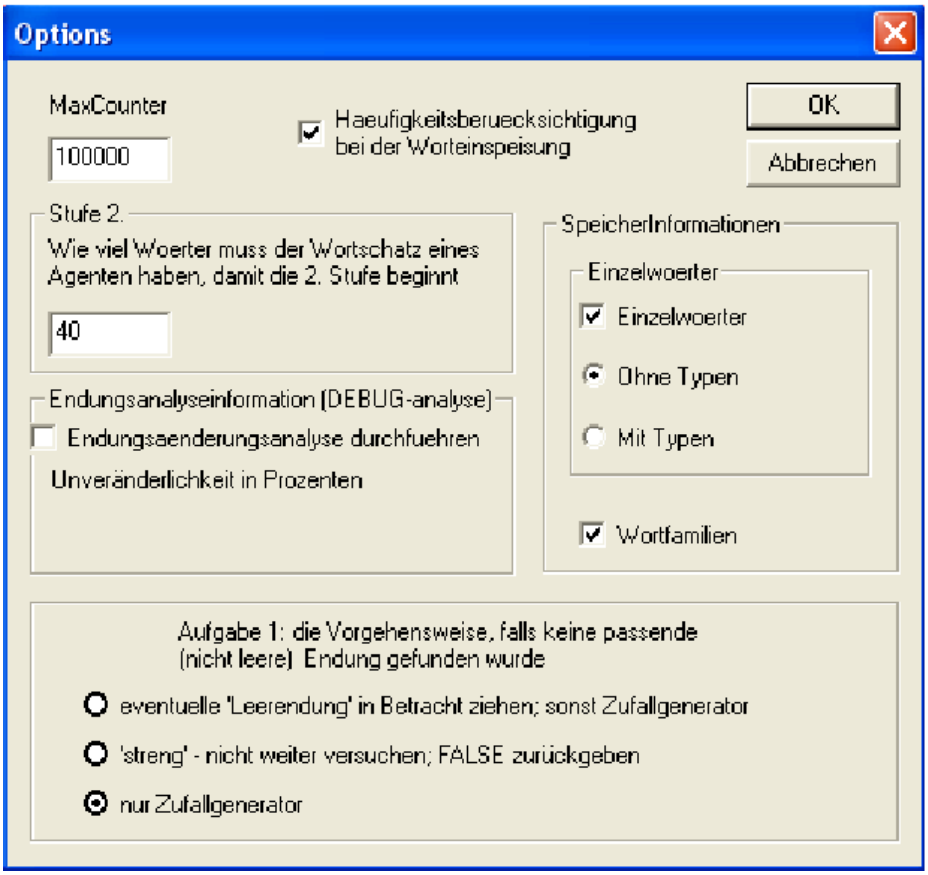

Figure 3.2: The options-dialog that allows to vary the parameters: number of iterations, frequency of words uttered to the agents, the amount of feedback, output options, etc. *Pustylnikov* (2010).

stage of the game (see the explanations below). Each child agent has to solve several communicative tasks during the iteration, and it gets scores for correctly guessing or producing words (see Fig. 3.3). In the following, we will refer to the single steps (i-iii) (see Fig. 3.3) to describe the stages of the game.

First Stage

A game round in this stage can be shortly summarized as follows:

1. A randomly selected receiver (child) has to guess the word class of the word produced by the adult by using its derivation network (i).

2. If 1. was successful, the child constructs a new word from another word that is present in its lexicon and a randomly chosen word class, for example, shine → shiny (ii). The word formation is checked by the adult (ii.i-ii.ii).

3. If 2. was successful, the child utters the word acquired in 1. to its sibling (iii).

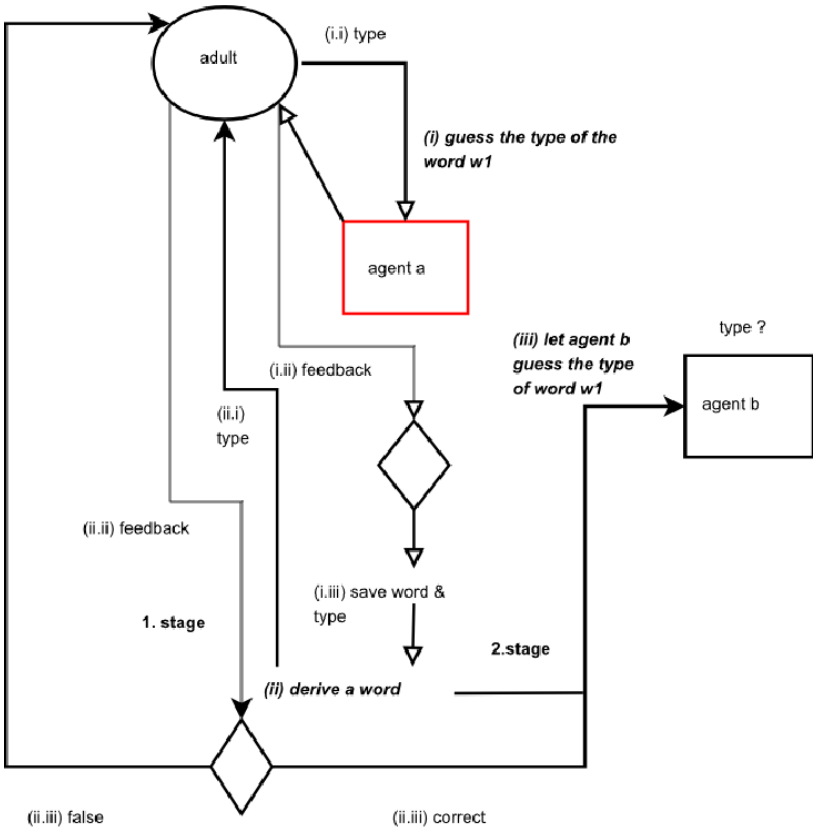

Figure 3.3: Overview: game procedure.

Second Stage

After the predefined number of words in the derivation network (option "`Stufe 2`" in Fig. 3.2) of one of the agents is reached, all the agents enter the second stage. In this stage the child perceives the words from the adult's lexicon. The lexicon contains more words than the derivation network, thus, for some words the possible derivations may be unknown even to the adult. A usual round in this stage can be described as follows:

1. The child has to guess the word class of the word received from the adult's lexicon **or** derivation network (i). The correctness of the answer is no longer controlled.

2. The child derives a new word from an existing word from its lexicon to a randomly chosen word class. Again, the answer is not validated by the adult (ii).

3. The child utters the word from 1) to their sibling (iii).

Note: the success of the guess is validated according to the speaker's knowledge about the word class. That is, if in 1. the speaker classified "*funny*" as an adjective, and now it utters it to the next agent, both will get a score, if the other recognizes *funny*

Word Class	English 49	German 49	German 421
NN	er	ung	ung
	tion	er	er
	le	en	en
VV	ss	en	en
	ry	ieren	ieren
	er	gen	ten
AA	al	er	bar
	tive	lich	lich
	ity	ig	ig

Table 3.1: Best suffixes induced via decomposition for the derivation networks of the size: 49 for English, 49 for German and 421 for German. The correct suffixes are underlined.

as an adjective too. Multiple forms, like the English *ease* (noun or verb) can be learned and differentiated by the agents.

In sum, in the second stage it is not obvious that children just inherit the knowledge learned from the adult. Rather, the communicative dynamics can force them to diverge from the learned behavior. In the second stage of our experiments we test how much knowledge is needed to produce the correct derivations in a particular language.

3.4 Experimentation

We test the system with respect to the following aspects: the decomposability of the language into stem and potential suffixes and the communicative success of the agents. The decomposability is evaluated by means of the *F-score*, which indicates the correctness of decomposed suffixes to suffixes of a particular language. The F-score averages on *precision* and *recall*, which are applied in information retrieval to measure the quality of a classification. Precision $\frac{\#\{correctly\ identified\ \cap\ identified\}}{\#\{identified\}} \in [0,1]$ relates the number of correctly identified suffixes to the total number of suffixes found by the system as representing a word class. Recall $\frac{\#\{correctly\ identified\ \cap\ identified\}}{\#\{correct\}} \in [0,1]$ relates the correctly identified suffixes to the total number of correct suffixes considered for a language. Then, the F-score can be calculated as follows: $F - score = \frac{2}{\frac{1}{recall_i} + \frac{1}{precision_i}} \in [0,1]$.[12] An F-score of 1 indicates a perfect match of real suffixes identified by the system, an F-score of 0 shows a complete mismatch, respectively.

To evaluate the communicative success of the game, we calculate the suffix ratio of all agents after a game run. For comparison and contrast to English and German, a third, random language is used that has no productive suffixes. We ask, whether productive rules emerge during communication for all of the three languages.

[12] *Hotho et al.* (2005).

language	#deriv.net	F-score	random baseline
German	49	0.77	0.66
English	49	0.55	0.66
German	421	0.88	0.66

Table 3.2: Suffix F-scores - suffixes identified by the adult analyzing the whole data versus real suffixes of a language.

Suffix Ratio

$$SR_{Ra1} = \frac{\sum_{a1 \neq a2, a1, a2 \in A} \sum_{c}^{C} \sum_{r=1}^{R} \sigma(s(a1, c, r), s(a2, c, r)) * \frac{1}{r}}{\sum_{r=1}^{R} \frac{1}{r} * C * N} \in [0, 1) \qquad (3.1)$$

The suffix ratio SR_{Ra1} is computed with respect to a particular agent $a1$ (adult or child). Suffixes stored for each word class C are ranked by the rank order R according to their success in the game. The suffixes s on the rank r of an agent (adult or child) $a1$ are compared to the suffix of an agent (adult or child) $a2$ on the respective rank. $\sigma(s(a1, c, r), s(a2, c, r))$ is a binary predicate that takes the value of 1 when two suffixes are equal and 0 otherwise. N is the number of agents.

The rank r is used as a weighting factor giving deviations on lower ranks a smaller value. This is because the first suffix is more likely to be used for production of new words than other suffixes. So, in our experiments we calculate SR_1 and consider SR_3 a complementary information resource, that is, we evaluate the number of successfully replicated suffixes on the first and on the first three ranks. In the denominator of Equation 3.1 the sum $\sum_{r=1}^{R} \frac{1}{r}$ gives the maximal value the sum $\sum_{r=1}^{R} \sigma(s(a1, c, r), s(a2, c, r)) * \frac{1}{r}$ can take when we assume that all suffixes coincide.

3.5 Results and Discussion

Table 3.1 and the F-scores in Table 3.2 show the three best suffixes for each word class identified by the system. The underlined suffixes are those that really exist in a language.[13] For German, the size of the input derivation network (49 vs. 421) does not play a crucial role for the decomposition's success. The network of 421 word families enhances the list solely by one more suffix (F-score 0.77 vs. 0.88). In English, only a small number of real suffixes is replicated. The F-score of 0.55 is even below the baseline for randomly assigning suffixes to word classes. This fact indicates that German words contain more information about the word class than English using productive suffixes for word formation.

[13]The system does not make a distinction between derivational and inflectional suffixes. This is because the word forms uttered to the agents appear in their infinitive form, so the identified suffixes are those that help to transform one word class to another, or more specifically, into its base form. This explains, why the German '-en' suffix is identified although it is grammatically classified as an inflectional infinitive suffix. This simplification allows us to test the decomposition algorithm in a minimal setting, however, it can be easily extended to learn more sophisticated inflectional relations.

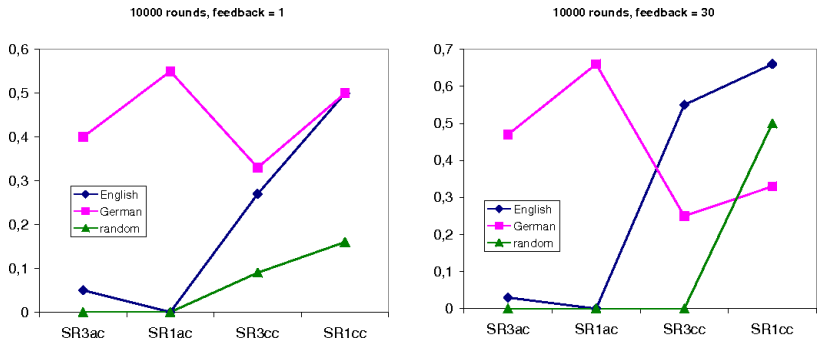

Figure 3.4: Suffix Ratio: SR_1, SR_3 were calculated for adult-children SR_{3ac}, SR_{1ac} and among children SR_{3cc}, SR_{1cc}.

However, this does not mean that the communication in English among the agents fails. Looking at suffixes after a communication round of the game for English, German and a random word set, we get the average suffix ratios (SR) displayed in Figure 3.4. We ran the game up to 10 times through 10,000 iterations and varied the amount of feedback given by the adult from 1 to 30. We observe that German has a high agreement among adult and children for both forms of feedback. This allows us to conclude that the amount of feedback does not influence the induction of correct derivation suffixes significantly. Children show less agreement but they consistently agree with the adult (perhaps on different suffixes).

English exhibits almost no agreement between adult and children but a high agreement among children. This is an interesting finding which is probably attributed to the general use of productive suffixes in English. Children decompose words using the same mechanism as the adult, and processing more words should normally lead to a convergence with the adult, if productive suffixes are present in the language. In the case of English, however, the resulting suffixes of the children are different from those of the adult which points to lower predictability of word classes based on suffixes. Thus, children learn the words as a whole, by the full word route and less via parsing. This leads to an overall success of the game, although, the derivation rules might be different from the adult's rules.

In terms of an agreement on common suffixes, a language emerges in all languages tested here. Even a completely random input produces a convergence of children's suffixes, which differ from the adult's random pseudo suffixes. Thus, structure emerges via communication, although, the structure induced by the adult may vary.

3.6 Summary

In this chapter we presented a system modeling the learning of derivation morphology. We found that structural status of the input words influences the newly emergent language.

Two languages, English and German, were compared by the amount of productive suffixes used for word formation. Although, substantial differences were observed,

the evolutionary game dynamics let the agents converge on a common language. The success of communication was observed in all cases, even for the random language. If the input language does not exhibit a clear structuring, children converge in the long run on common suffixes. However, the resulting language might be different from that of the adult.

The explicit learning modeled in the first stage does not play a crucial role for the emergence of a structured input. It rather can encourage the children to learn correct word to word class relations. The structure of the lexical input determines the persistence of the given language, to a large extent. If no structural regularities can be induced from the language of the adult, the language is less stable, and in most cases, children have to negotiate other rules facilitating the communication.

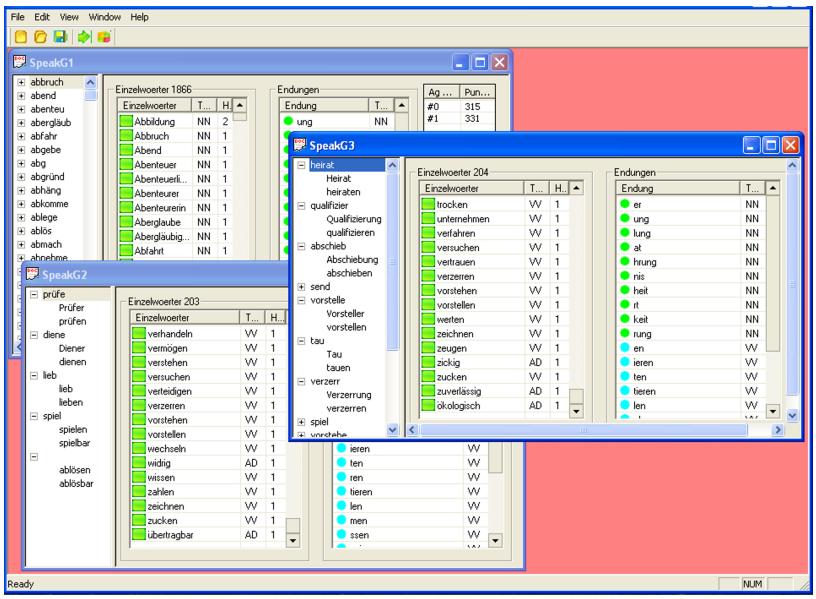

Figure 3.5: The Figure shows the system at the beginning of the game. The top-left window represents the adult, the other windows represent children. The small amount of common suffixes (third column) and the very few word families (first column) show that a common language has not been developed yet.

CHAPTER IV

Morphological Networks

4.1 Introduction

Network models have gained importance in the humanities in recent years. A recently emerging branch of interdisciplinary research interest complements linguistic studies with methods in network theory. Following *Köhler* (1986) a language is a complex dynamic system built of highly structured components (linguistic levels like syntax, morphology, etc.) that influence each other as well as they are influenced by other linguistic and non-linguistic factors. Standard approaches in linguistics enable to describe single linguistic phenomena or cross-linguistic patterns (language universals) precisely leaving aside the complex interactions between linguistic units in total. Quantitative linguistics (see e.g., *Altmann and Lehfeldt* (1973); *Köhler* (1986)) bridges between linguistic phenomena, on the one hand, and their relations, on the other, using quantitative methods. Network models of language were recently discovered as an appropriate means to study the organization principles of language quantitatively (*Ferrer i Cancho et al.*, 2004; *Mehler et al.*, 2010a; *Liu and Xu*, 2011). The reason is that networks allow to represent complex relations between linguistic units, allowing to "zoom in", and inspect their regularities.

Ferrer i Cancho et al. (2007) have shown that syntactic dependency networks of 6 different languages exhibit the same *small-world* (*Watts and Strogatz*, 1998) property, which seems to be universal for languages. *Mehler* (2008a) found out that ontological and functional differences of Wiki-networks can be recovered in examining their topology. *Abramov and Mehler* (2011) used the same network model to cluster 11 languages into 3 genetic groups. *Liu* (2008) could distinguish among genres within a single language comparing syntactic dependency networks of the same language. *Mehler et al.* (2010a) presented a novel approach to the Sapir-Whorf Hypothesis analyzing networks based on social ontologies of Wikipedia. *Mehler et al.* (2010b) introduce a network model of dialog based on lexicons of communication partners, and demonstrate its potential for predicting lexical alignment of interlocutors. These and other studies on networks suggest that we can enhance linguistic studies by means of network analysis.

The study presented in this chapter aims to complement this field of research by presenting a network model for derivational morphology. We apply information-

theoretic measures to study the properties of morphological derivation networks in comparison with to random graphs. The goal of the chapter is twofold: to promote the use of network models in linguistics, and to evaluate some information theoretic measures for different kinds of graphs. We proceed as follows:

- In particular, we focus on suffixation as one mechanism to derive new words from existing ones. We construct a network from words, derivation suffixes and parts of speech (PoS) as well as the relations between them.
- In measuring the entropy of these networks by means of so called *information functionals* (see Sec. 4.3 for definition) we aim to capture the variation between typologically different languages. In this way, we rely on the work of *Dehmer* (2008) who has recently introduced a framework for the measurement of the entropy of graphs.
- Additionally, we examine to check whether the entropy measures allow us to distinguish between language networks, on the one hand, and random networks, on the other. In doing so, we rely on the work of *Mehler* (2008a, 2009), that is, *Quantitative Network Analysis* (QNA) as a framework of network classification.

To the best of our knowledge this is the first empirically founded network model of morphology. It brings together two research branches,QNA and graph entropy measurement, in order to shed light on an area of linguistic networking whose cognitive relevance has recently been claimed by *Bybee* (1988).

Section 4.2 explains how the networks were induced, gives their formal definition, and discusses some of their properties. In Section 4.3, we present and discuss different graph entropy measures based on information functionals. In this Section, we provide some modifications on the approach of *Dehmer* (2008) adapting it for our purpose. We evaluate these functionals based on some characteristic example graphs (Sec. 4.4). In Section 4.5, we compute these functionals for language networks and compare them to random graphs. We cluster the graphs based on the values of the functionals. The results show that language networks can be distinguished perfectly from the random ones using the functionals. In Section 4.6, we discuss the results. We summarize our findings in Section 4.7.

4.2 Morphological Derivation Networks

The notion of the *morphological derivation network* (MDN) introduced here is attributed to the organization of linguistic units in the area of derivational morphology.

In this section we describe how the networks were obtained (Sec. 4.2.1) and present their formal definition (Sec. 4.2.2).

4.2.1 Decomposition of productive suffixes

In this section, we briefly recapitulate the decomposition algorithm (henceforth referred to as the *decomposition algorithm* DA) presented in Chapter III that is responsible for the induction of derivation rules from lexical input. The underlying

theoretical framework behind DA is based on models of morphological processing (see *Dressler and Karpf* (1995); *Bertinetto and Noccetti* (2006); *Clahsen et al.* (2003)). In these models, suffixes that have the same function (e.g., to derive an adjective from a noun) are supposed to compete during the evolution of language. For example, a suffix that is preferably used to derive an adjective from a noun is likely to be reused in future word formations (e.g., *fruit* $\rightarrow$ *fruit-ful*). These suffixes are called productive suffixes of a language (*Baayen*, 1991).

DA detects derivation suffixes in a language (if such exist) decomposing words into suffixes and stems. *Pustylnikov and Schneider-Wiejowski* (2009) could show that DA is able to identify productive suffixes in German through analyzing texts from different periods of time (17-19th vs. 20th century) and different registers (i.e., spoken vs. written). In this chapter, we construct the MDNs using the output of DA (i.e., suffixes and stems). The procedure underlying DA can be summarized as follows:

DA parses texts that are pre-tagged with PoS information, that is, where the word category of each word is given. Four filtering steps are applied to filter out the derivation suffixes used in the input language (see Chapter III for details). Roughly speaking, suffixes found in combination with a large number of different stems forming a particular PoS are considered to be *significant* in language. The ten most significant suffixes are detected for each PoS that are most likely to to form new words. These suffixes (as well as the corresponding stems) are taken to construct the MDNs.

4.2.2 Network Definition

In the previous section we outlined the decomposition algorithm (DA) which is used here to induce the derivation networks. Formally, the MDNs are multi-level graphs (see *Mehler* (2008a)) partitioned into three disjunct subsets of vertices, that is, three-level graphs.

Definition 1. *Let $G = (V, E)$ be a graph of vertices V and edges E. We call G a three-level graph if the set V is represented as a union of three non-overlapping subsets, that is $V = V_1 \cup V_2 \cup V_3$ with $V_i \cap V_j = \emptyset, \forall i, j = 1, 2, 3 \wedge i \neq j$. There can exist edges between vertices of the three subsets of V as well as between vertices of particular subsets.*

The vertex subsets of V are obtained from three different sources that are described below. Instead of utilizing the subsets V_1, V_2 and V_3, we will utilize W, S and P respectively, which are explained in the following.

1. Vertices belonging to the first subset W are **words** and stems obtained from the lexical input.

2. The second subset S contains significant **suffixes** identified via decomposition (see the previous section).

3. The last subset P includes **PoS**.

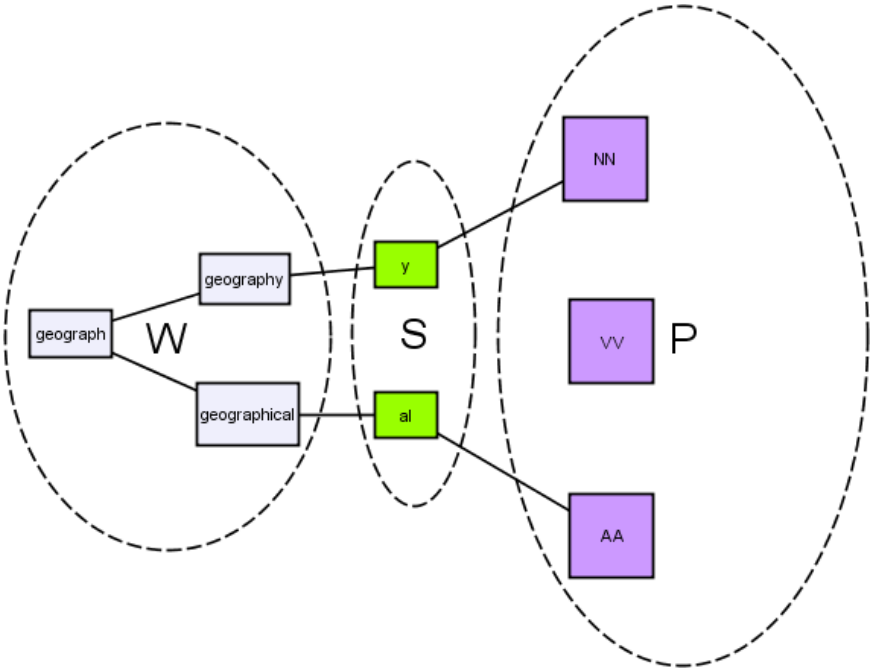

Figure 4.1: An example MDN. Subsets W = words and stems, S = suffixes and P = PoS.

Note that Definition 1 differs from the definition of a k-partite graph, which does not allow for edges between vertices within a subset V_i. In our case, edges can occur between different subsets (e.g., between S and P) as well as within the set W. No edges occur among vertices of the subsets S and P. These observations restrict the connectivity of the graph G (see also Fig. 4.1 for illustration) in the following way.

1. If $v \in W, w \in W$ an edge $\langle v, w \rangle \in E$ can exist.
2. If $\{v, w\} \in S \rightarrow \langle v, w \rangle \notin E$.
3. If $\{v, w\} \in P \rightarrow \langle v, w \rangle \notin E$.

This network model allows us to map the morphological structure of different languages (i.e. English vs. German and vs. a language without derivational morphology). For example, in a language that does not use suffixes the second level S will be missing.

4.2.3 Data: Networks and their Topological Properties

In this chapter we compare three kinds of graphs:

1. morphological derivation networks (MDN):
   - English
   - German

- Random Words MDN (Random Morphological Derivation Network (RMDN))

2. random graphs:[1]
   - *Erdős and Rényi* graphs (ERN)
3. small world graphs:
   - scale-free graphs (*Barabási and Albert*, 1999)
   - small world graphs (*Watts and Strogatz*, 1998)

The MDNs were constructed based on word lists of the same size as those that were decomposed into stems and suffixes using DA. ERNs are randomly generated networks with cardinalities of German and English MDNs. The RMDN was constructed from a randomly generated word list of nonsense words that consist of letters from the Latin alphabet. This word list served as input to DA. The DA tried to induce significant derivation suffixes from random words. Of course, the nonsense words did not have any internal structure, though, some suffixes occurring by chance in more than one word were detected. This fact explains some coincidental links between the noun and verb subgraphs in Figure 4.7. As can be seen, single words and suffixes group around the three PoS categories without any additional organization within the word families.

The English and German MDNs are displayed in Figures 4.5 and 4.6. Table 4.1 lists some of their topological characteristics.[2] In the following, we will focus on some of their properties.

When we compare the connectivity of both language networks to RMDN, we see that the former two have many more edges than vertices. The proportion of vertices to edges is also similar for both natural networks. The RMDN has more vertices than edges and two disconnected components. This can be explained by a very small number of common parts (i.e., stems, suffixes) in the random vocabulary. These random words have a low average degree (0.98) by connecting to PoS vertices.[3] The RMDN has three parts of vertices that group together (star-like) around the three main PoS. The noun and verb parts are connected together by a coincidental link; adjectives constitute a separate component. All in all, the star-graph-like shape of the RMDN is also confirmed by the higher centralization (*Freeman*, 1978-1979) (3.5 vs. ~ 2) and heterogeneity (*Snijders*, 1981) (3.08 vs. ~ 1.4) values in contrast to natural networks.

Density (*Snijders*, 1981) is a parameter that indicates the average number of neighbors. Density ranges from $[0, 1]$ and shows how densely the vertices of a graph

[1] All random graphs, small world graphs excluding RMDNs were generated by Cytoscape (`www.cytoscape.org`), a tool for network analysis. ER graphs are connected undirected random *Erdős and Rényi* graphs of the cardinality of German and English. BA *Barabási and Albert* (1999) and WA *Watts and Strogatz* (1998) are randomly generated small world graphs of the cardinality of German.

[2] The topological network characteristics for all graphs were computed by means of Network Analyzer (`http://med.bioinf.mpi-inf.mpg.de/netanalyzer/`), a module for Cytoscape.

[3] In fact, only 4 vertices of 136 have a degree > 1.

Feature	German	English	RMDN	ERN 195	ERN 163	BA 195	WS 195
#Nodes	195	163	136	195	163	195	195
#Edges	297	255	134	939	611	387	585
#Self Loops	4	22	0	10	5	0	128
#Con. Compon.	1	1	2	1	1	1	1
Diameter	7	7	7	5	5	6	7
Radius	4	4	1	3	4	4	5
Av.shortest path len.	3.66	3.68	3.99	2.58	2.76	3.31	3.84
Clustering Coef.	0.01	0.08	0	0.04	0.05	0.087	0.15
Density	0.015	0.018	0.015	0.04	0.04	0.02	0.02
Heterogeneity	1.32	1.34	3.08	0.3	0.3	1.08	0.27
Centralization	0.19	0.21	0.354	0.04	0.03	0.17	0.02
Av. degree	1.51	1.49	0.98	10.26	11.64	1.98	3
$\gamma(degree)$	-1.91	-1.98	-3.09	-0.63	-0.61	-1.6	-0.06
$R^2(degree)$	0.98	0.99	0.99	0.69	0.81	0.99	0.96
Av. BC	0.014	0.037	0.02	0.008	0.01	0.02	0.02
$\gamma(BC)$	-0.99	-2.29	-1.01	-0.35	-0.39	-0.85	-0.05
$R^2(BC)$	0.99	0.99	0.77	0.84	0.81	0.96	0.99

Table 4.1: Network Characteristics: number of vertices, edges, self loops, connected components, diameter, radius, average shortest path length, clustering coefficient (*Watts and Strogatz*, 1998), density (*Snijders*, 1981), heterogeneity (*Snijders*, 1981), centralization (*Freeman*, 1978-1979), average degree, gamma and the coefficient of determination of the power law fit of degrees, average betweenness centrality (BC) (*Brandes*, 2001), gamma and the coefficient of determination of the power law fit of BCs.

are connected. For all the MDNs, the density values are comparably small as are the values of the clustering coefficient (*Watts and Strogatz*, 1998). This can be explained by a selective connectivity among the three levels of the multi-level graph, as explained in Section 4.2.2. In all MDNs, vertices of the set P do not link together but do link to the vertices of S and W. Vertices of W do not connect randomly to one another since word families have a limited size. The same holds for suffixes, which do not connect together within S but connect only to W and P. These peculiarities lower the probability of common neighbors in the MDN, for random as well as natural networks. Note that English has a slightly higher density value (0.018) and a slightly higher clustering coefficient (0.08) than the other networks. This can be attributed to the English language's reduced morphological variety within the word, which leads to a multi-functionality of items in W, and raises their probability to be connected to each other. If the words in W were completely random, the lack of significant suffixes would lead to a structure comparable to RMDN. However, English words are not just random combinations of letters that result in a redundancy of the same words in several classes and a higher connectedness among them.

A remarkable property is the number of self loops showing different values for all the three MDNs. English has a remarkably higher number of self loops than German and RMDN ($22 > 4 > 0$). This property also points to the fact that the same word forms in English often function as different PoS and stems. For example, the word *lift* can occur as a stem, verb and noun. English contains many similar examples, thus, the number of self loops discriminates English from other networks according to morphological property.

In summary, the MDNs of natural languages can be distinguished from the random MDNs when comparing their topological properties. Furthermore, differences in morphological structure among natural languages become visible by looking at the topology of MDNs. In Figures 4.5, 4.6 and 4.7 the different MDNs are visualized by their centrality values (i.e., the vertex with the highest centrality is the largest one). In English and in RMDN the central vertices are *nouns*, *verbs* and *adjectives*. In German, in contrast, the most central are *nouns*, the suffix *-en* and *adjectives*. In this network, *-en* is more central than the PoS VV ('verb') because this suffix is attached not only to almost all the verbs in the German infinitive, but also to a large number of nouns (*seh-en* vs. *das Seh-en*). This example illustrates how a particular language's morphological differences form the topology of the network.

4.3 Measuring the Entropy of MDNs

In this section we present and discuss several information functionals that we evaluate for our networks.

4.3.1 Graph Entropy by means of Information Functionals

We build on the approach of graph entropy measurement as developed by *Dehmer* (2008). Instead of determining partitions, involving a graph invariant which may turn out computationally costly, he relies on assigning a probability value to each vertex of a graph. This is what we need when evaluating the structural role of single morphemes in a morphological network.

This section recapitulates basic notions of graph entropy measurement as introduced by *Dehmer* (2008). This holds for Definitions 2-6, which basically repeat the corresponding framework of *Dehmer* (2008). We complement several propositions as well as a lemma on information functionals as the foundation stones of entropy measurement. This will be the starting point of measuring the entropy of morphological networks as proposed in the following sections.

Following Definition 2.7 in *Dehmer* (2008), if we have a finite undirected graph $G = (V, E)$ with V being the set of vertices and E the set of edges, and we have a positive function f on the set V (called an *information functional* in *Dehmer* (2008)), then we can define:

$$p(v_i) = \frac{f(v_i)}{\sum_{k=1}^{|V|} f(v_k)}. \tag{4.1}$$

Since the equality $p(v_1) + p(v_2) + ... + p(v_{|V|}) = 1$ holds, we can interpret the values $p(v_i)$ $(i = 1, 2, ..., |V|)$ as vertex probabilities.

Having such a probability distribution, we immediately compute the entropy $I_f(v)$ of G, which is interpreted here as the mean structural information content.[4]

$$I_f(G) = -\sum_{i=1}^{|V|} p(v_i) \ln p(v_i). \tag{4.2}$$

Dehmer (2008) presents some novel information functionals of V which capture, in some sense, the structural information of the underlying graph G. We concentrate on the functional f^V and prove some statements (in Propositions IV.2-3, Lemma IV.1), which allow us simplify to use f^V for our purpose.

We first need to repeat some preliminary definitions as well as the definitions of the information functional f^V given in *Dehmer* (2008). Note that the length of a path on the graph G is measured (in *Dehmer* (2008) and here) as a number of edges in this path. We denote $\forall u, v \in V$ the length of the shortest path between them by $d(u, v)$.

Definition 2. *The quantity $\rho = \rho(G) := max_{u,v \in V} d(u, v)$ is called the diameter of G.*

Definition 3. *The set $S_j(v_i, G) := \{v \in V | d(v_i, v) = j, j \geq 1\}$ is called the j-sphere of v_i regarding G.*

Definition 4. *Given a vertex $v_i \in V$ and the j-sphere $S_j(v_i, G)$, according to Dehmer (2008), we define the local information graph $\mathcal{L}_G(v_i, j)$ as follows: for all $w \in S_j(v_i, G)$ the shortest path connecting w and v_i has the length j by definition of $S_j(v_i, G)$. There is not necessarily only one such path for the vertex w, but we take only one path of the length j for every $w \in S_j(v_i, G)$. Then, these paths with their edges and vertices form a subgraph of G which is called the local information graph and denoted by $\mathcal{L}_G(v_i, j)$. j is called the local information radius regarding v_i.*

Now we formulate and prove a lemma which shows that $\forall v_i \in V$ all j-spheres $S_j(v_i, G)$ with $j = 1, 2, ..., \rho$ cover the set $V \setminus \{v_i\}$ and $\forall j, k$ with $1 \leq j, k \leq \rho$ and $k \neq j$ the equality $S_j(v_i, G) \cap S_k(v_i, G) = \emptyset$ holds.

Lemma IV.1. *Let $G = (V, E)$ be a finite undirected connected graph. Then $\forall v_i \in V$ the equality $\sum_{j=1}^{\rho} |S_j(v_i, G)| = |V| - 1$ holds where $|V|$ $(|S_j(v_i, G)|)$ is the cardinality of the set V $(S_j(v_i, G))$ respectively, and $\rho = \rho(G)$ is the diameter of G as defined above.*

Proof. Let v_i be an arbitrary vertex on G. We show that $\forall w \in V \setminus \{v_i\}$ there exists a *j-sphere* $S_j(v_i, G)$ on which the vertex lies. Indeed, if we take $j = d(w, v_i)$ then, obviously, $w \in S_j(v_i, G)$. Furthermore, if we have two natural numbers k and j with $k \neq j$ and $j \leq k, j \leq \rho$ we can easily see that

$$S_j(v_i, G) \cap S_k(v_i, G) = \emptyset.$$

These two observations complete the proof of Lemma IV.1. □ □

[4] *Dehmer* (2008) uses log to calculate the entropy. We use ln here for all functionals, which does not have any impact on the final results of the relative entropy (see the definition below) values.

Definition 5. *Given a local information graph $\mathcal{L}_G(v_i, j)$ regarding $v_i \in V$ we denote (see Dehmer (2008)) the sum of the lengths of all shortest paths in $\mathcal{L}_G(v_i, j)$ selected in Definition 4, each of which connects v_i with some point of $S_j(v_i, G)$ by $l(P(\mathcal{L}_G(v_i, j)))$.*

It was proved in *Dehmer* (2008) (see Proposition 3.1) that $\forall v_i \in V$ and $j = 1, 2, ..., \rho$ it holds:

$$l(P(\mathcal{L}_G(v_i, j))) = j|S_j(v_i, G)| \tag{4.3}$$

We are now able to present the definition of the information functional f^V introduced in *Dehmer* (2008).

Definition 6. *The information functional f^V is defined $\forall v_i \in V$ by the formula:*

$$f^V(v_i) := \alpha^{\sum_{j=1}^{\rho} c_j |S_j(v_i,G)|} \tag{4.4}$$

where c_j with $j = 1, 2, ..., \rho$ and α are arbitrary real positive parameters.

In the following, we prove some properties of f^V.

Proposition IV.2. *If $c_1 = c_2 = ... = c_\rho > 0$ then $\forall \alpha > 0, \forall v_i, v_j \in V$ we have*

$$f^V(v_i) = f^V(v_j). \tag{4.5}$$

Proof. In view of our assumptions and Lemma IV.1 $\forall v_i \in V$ we have:

$$\begin{aligned} f^V(v_i) &= \alpha^{\sum_{j=1}^{\rho} c_j |S_j(v_i,G)|} \\ &= \alpha^{c_1 * \sum_{j=1}^{\rho} |S_j(v_i,G)|} \\ &= \alpha^{c_1(|V|-1)}. \end{aligned}$$

So, the value $f^V(v_i)$ does not depend on the vertex v_i. □ □

Corollary IV.3. *If $c_1 = c_2 = ... = c_\rho > 0$ then for the probability distribution p^V on V induced by the information functional f^V with the formula (4.1), $\forall \alpha > 0, \forall v_i, v_j \in V$ we have $p^V(v_i) = p^V(v_j)$. The corresponding entropy has the maximal possible value for G which equals $\ln |V|$ where $|V|$ is the cardinality of the vertex set V.*

Proof. In view of (4.1) and Proposition IV.2 $\forall v_i, v_j \in V$ we immediately obtain:

$$\begin{aligned} p^V(v_i) &= p^V(v_j) \\ &= \frac{1}{|V|} \end{aligned}$$

and for the entropy $I_{f^V}(G)$ we get:

$$\begin{aligned} I_{f^V}(G) &= -\sum_{i=1}^{|V|} \frac{1}{|V|} * \ln\left(\frac{1}{|V|}\right) \\ &= \ln(|V|) \end{aligned}$$

□

Proposition IV.4. *Given a set of positive parameters $\alpha, c_1, c_2, ..., c_\rho$ for the information functional f^V and an arbitrary positive number c, we consider another set of parameters $c_1 + c, c_2 + c, ..., c_\rho + c$ with the same α. Then the probability distribution and hence the entropy for both parameter sets are equal.*

Proof. To prove the statement of Proposition IV.4, we denote $\forall v_i \in V$ the values of f^V with parameters $c_1, c_2, ..., c_\rho$ by $f^V(v_i, c_1, ..., c_\rho)$. We have

$$\begin{aligned} f^V(v_i, c_1 + c, c_2 + c, ..., c_\rho + c) &= \alpha^{\sum_{j=1}^{\rho}(c_j+c)*|S_j(v_i,G)|} \\ &= \alpha^{\sum_{j=1}^{\rho} c_j*|S_j(v_i,G)|+c*\sum_{j=1}^{\rho}|S_j(v_i,G)|}. \end{aligned}$$

In view of Lemma IV.1 we obtain

$$f^V(v_i, c_1 + c, c_2 + c, ..., c_\rho + c) = f^V(v_i, c_1, c_2, ..., c_\rho) * \alpha^{c(|V|-1)}.$$

Thus, we can see that if we add a constant $c > 0$ to each $c_j, j = 1, 2, ..., \rho$ $\forall v_i \in V$ the new value of f^V will be a product of the old value and the constant $\alpha^{c(|V|-1)}$. The corresponding probability distribution, and hence, the entropy value does not change, as it ensures from Equation (4.1). □

Proposition IV.5. *If $\alpha > 1$ ($\alpha < 1$) for f^V we can set $\alpha = 2$ ($\alpha = \frac{1}{2}$) respectively without loss of generality.*

Proof. Given a set of positive parameters $\alpha > 1, c_1, c_2, ..., c_\rho$ we can choose the parameters $c'_1, c'_2, ..., c'_\rho$, so that $\forall v_i \in V$

$$\alpha^{\sum_{j=1}^{\rho} c_j*|S_j(v_i,G)|} = 2^{\sum_{j=1}^{\rho} c'_j*|S_j(v_i,G)|}$$

holds. Indeed, if we put $c'_j = c_j * \log_2 \alpha, j = 1, 2, ..., \rho$ then we get

$$\begin{aligned} 2^{\sum_{j=1}^{\rho} c'_j*|S_j(v_i,G)|} &= 2^{(\sum_{j=1}^{\rho} c_j*|S_j(v_i,G)|)*\log_2 \alpha} \\ &= \alpha^{\sum_{j=1}^{\rho} c_j*|S_j(v_i,G)|} \end{aligned}$$

□

So, we see that if we have a set $\alpha, c_1, c_2, ..., c_\rho$ of positive parameters with $\alpha > 1$ and consider the other set of positive parameters $2, c_1 * \log_2 \alpha, c_2 * \log_2 \alpha, ..., c_\rho * \log_2 \alpha$, then $\forall v_i \in V$ the value of the information functional f^V does not change.

Remark IV.6. Proposition IV.5 shows that we can reduce the number of parameters for the information functional f^V by taking $\alpha = 2$ (or $\alpha = \frac{1}{2}$). So, f^V can be now defined by the following equation:

$$f^V(v_i) := 2^{\sum_{j=1}^{\rho} c_j*|S_j(v_i,G)|}, \forall v_i \in V \tag{4.6}$$

with $c_1, c_2, ..., c_\rho$ being positive parameters.

Moreover, if we consider the numbers $c_1, c_2, ..., c_\rho$ being simultaneously positive or negative, we can cover both the cases $\alpha = 2$ and $\alpha = \frac{1}{2}$ with the formula above. In short, we can treat the set $c_1, c_2, ..., c_\rho$ of parameters as a ρ-dimensional vector $\bar{c} = \{c_1, c_2, ..., c_\rho\}$ and the set $|S_1(v_i, G)|$,$|S_2(v_i, G)|$, ...,
$|S_\rho(v_i, G)|$ $\forall v_i \in V$ as the vector function $\bar{S}(v_i) = \{|S_1(v_i, G)|, |S_2(v_i, G)|, ..., |$
$S_\rho(v_i, G)|\}$. Furthermore, instead of dealing with the sum $\sum_{j=1}^{\rho} c_j |S_j(v_i, G)|$ we can write the scalar product $(\bar{c}, \bar{S}(v_i))$ of two ρ-dimensional vectors $\bar{c}$ and $\bar{S}(v_i))$, which is simply the sum $c_1 * |S_1(v_i, G)| + c_2 * |S_2(v_i, G)| + ... + c_\rho * |S_\rho(v_i, G)|$.

The formula which defines f^V can be given now as follows (*Dehmer et al.*, 2009):

$$f^V(v_i) = f^V(v_i, \bar{c}) := 2^{(\bar{c}, \bar{S}(v_i))} \tag{4.7}$$

where the coordinates of the ρ-dimensional vector $\bar{c}$ can be defined as all positive or all negative. So we see that it suffices to use the functional f^V, varying only one set of parameters $\{c_1, c_2, ..., c_\rho\}$ without any loss of information.

Remark IV.7. For simplicity, *Dehmer et al.* (2009) consider only the exponent in the formula 4.7. So, instead of using Equation 4.4 we can take the functional f^V as follows:

$$f^V(v_i) = (\bar{c}, \bar{S}(v_i)) \tag{4.8}$$

In the following, we will use this version of the functional when computing the entropy of the graphs.

4.3.2 Information Functional on the Set $J = \{1, 2, ..., \rho\}$

In this section we present the information functional f^J (*Konstantinova*, 2006) that is actually a function on the set $J = \{1, 2, ..., \rho\}$ with ρ being the diameter of the graph $G = (V, E)$.

Definition 7. *Using the Definition 3 of a j-sphere we define a function f^J on the set J as follows:*

$$f^J(j) := \sum_{i=1}^{|V|} |S_j(v_i, G)| \tag{4.9}$$

The value of $f^J(j)$ gives the sum of the cardinalities of all j-spheres in G. The probability $p^J(j)$ for j can be calculated by the standard formula:

$$p^J(j) = \frac{f^J(j)}{\sum_{i=1}^{\rho} f^J(i)}. \tag{4.10}$$

So, the entropy of G based on f^J can be calculated, as usual, according to the well-known formula of entropy as it was shown in the previous section (see also *Dehmer* (2008)).

$$I_f^J(G) = -\sum_{j=1}^{\rho} p^J(j) * \ln\left(p^J(j)\right) \tag{4.11}$$

In addition, we calculate the *relative entropy* of a graph given by the formula:

$$\bar{I}_f^J(G) = \frac{I(f^J)}{\ln \rho} \in [0,1] \tag{4.12}$$

for $\rho > 1$. For the functional f^V, respectively, we calculate

$$\bar{I}_f^V(G) = \frac{I(f^V)}{\ln |V|} \in [0,1] \tag{4.13}$$

with $|V|$ being the number of vertices in the graph.

4.3.3 Information Functional based on Distances

The next measure we like to present is the information functional based on distances in the graph proposed by *Konstantinova and Paleev* (1990) and evaluated in *Konstantinova* (2006) on molecular graphs. The results in *Konstantinova* (2006) state that this functional distinguishes well between polycyclic graphs and trees. Graphs studied in *Konstantinova* (2006) are small graphs representing molecular structures. In this chapter we test the ability of this functional to discriminate between more complex networks.

The information functional proposed in *Konstantinova* (2006) is calculated for a vertex v_i as the entropy of its shortest distances from all other vertices in the graph:

$$H_D(v_i) = -\sum_{u \in V} \frac{d(v_i,u)}{D(v_i)} \ln \frac{d(v_i,u)}{D(v_i)} \tag{4.14}$$

with $D(v_i) = \sum_{u \in V} d(v_i,u)$. The aggregation function across all distances of vertices in the graph is proposed in *Konstantinova* (2006) as follows:

$$H_D^V = \sum_{v \in V} H_D(v) \tag{4.15}$$

The codomain of this function does not lie within the interval of $[0,1]$, which is preferable in order to compare the graphs. Instead of normalizing the above function, we utilize the function $D(v_i)$ as an information functional:

$$f^D(v_i) := D(v_i) = \sum_{u \in V} d(v_i,u) \tag{4.16}$$

Then, the corresponding probability p^D is given by the formula:

$$p^D(v_i) = \frac{f^D(v_i)}{\sum_{v \in V} f^D(v)} \tag{4.17}$$

Given these probabilities, the entropy and the relative entropy can be calculated subsequently:

$$I_f^D(G) = -\sum_{v \in V} p^D(v) * \ln \left(p^D(v)\right) \tag{4.18}$$

$$\bar{I}_f^D(G) = \frac{I(f^D)}{\ln |V|} \in [0,1] \tag{4.19}$$

4.3.4 Information Functional based on the Distribution of Distance Sums

In addition to f^D, we present an information functional based on the distribution of distance sums. The reason for introducing this measure is the rather unsatisfactory separability of the functional f^D. *Konstantinova* (2006) shows that f^D possesses a high discriminative potential for distinguishing molecular graphs. However, molecular graphs used in *Konstantinova* (2006) are rather small graphs of about 20 vertices. Applying the functional to more complex networks results in a poor performance (see Tab. 4.7 in Sec. 4.5).

The reason for that might be that the sum $\sum_{v \in V} f^D(v)$ in Equation 4.17 produces some redundancy, since each distance sum $f^D(v_i)$ contains $|V| - 1$ other sums of other vertices (while the graph is connected). For our graphs, which are more complex than molecular graphs, the difference between single $f^D(v_i)$ and the total sum $\sum_{v \in V} f^D(v)$ is always large, resulting in similarly small probabilities $p^D(v_i)$ and in an indistinctive measure of entropy (see Tab. 4.7).

To overcome this problem, it was necessary to find another way to obtain the probabilities of distance sums. For this reason, we decided to explore the distribution of vertex sums by means of a new information functional.

We consider the functional f^{DS} on the set $\{1, 2, ..., R\}$ with R being the number of different values of the functional f^D on G (see Equation 4.16), that is, we enumerate somehow the different values of f^D on G using the numbers $1, 2, ..., R$. So, for each vertex $v \in V$ we get a number $\mathrm{ind}(v) \in \{1, 2, ..., R\}$ that equals to the number the value $f^D(v)$ has got by our enumeration. Thus, for any $v, u \in V$ the equality $\mathrm{ind}(v) = \mathrm{ind}(u)$ holds iff $f^D(v) = f^D(u)$. The functional f^{DS} can be defined as follows:

$$f^{DS}(k) := |\{v | v \in V, k = \mathrm{ind}(v)\}| \tag{4.20}$$

Now, the probability for each $k \in \{1, 2, ..., R\}$ can be defined subsequently:

$$p^{DS}(k) = \frac{f^{DS}(k)}{\sum_{i=1}^{R} f^{DS}(i)} = \frac{f^{DS}(k)}{|V|}. \tag{4.21}$$

The entropy and the relative entropy are calculated by the following formulae:

$$I_f^{DS}(G) = -\sum_{k=1}^{R} p^{DS}(k) * \ln\left(p^{DS}(k)\right) \tag{4.22}$$

$$\bar{I}_f^{DS}(G) = \frac{I(f^{DS})}{\ln R} \in [0, 1] \tag{4.23}$$

for $R > 1$.

4.3.5 Information Functional based on Betweenness Centralities

At least, we calculate the entropy based on the distribution of betweenness centralities (*Brandes*, 2001) (BC for short) of vertices in G. For each vertex $v \in V$ in

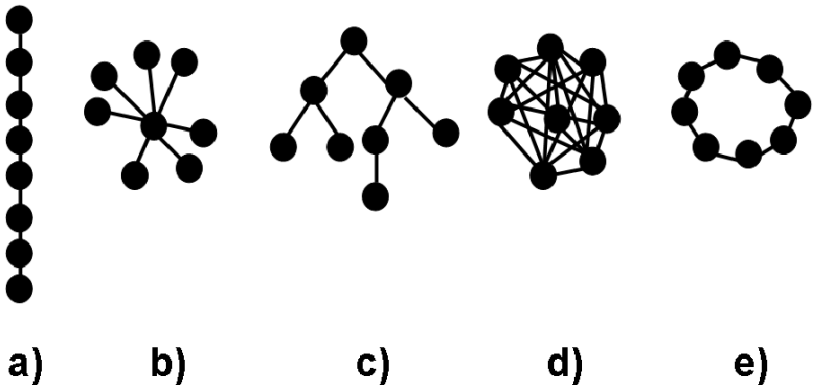

Figure 4.2: Example graphs of 8 vertices: a) linear graph, b) star graph, c) tree graph, d) complete graph (CG) and e) circle graph. The figure is taken from *Mehler* (2008a).

G, we calculate first the value $\mathrm{BC}(v)$ (see *Brandes* (2001)). Let l be the number of different values of the function BC. Then, we enumerate arbitrarily the various values of BC in G using the numbers $1, 2, ..., l$. So, for each vertex v we get an index $ind(v)$ that equals to the number the value $\mathrm{BC}(v)$ has, according to our enumeration. For any two vertices u and v the equality $\mathrm{ind}(v) = \mathrm{ind}(u)$ holds iff $\mathrm{BC}(v) = \mathrm{BC}(u)$.

Now we define the functional f^{BC} on the set $B = \{1, 2, ..., l\}$ $\forall k \in B$ as follows:

$$f^{BC}(k) := |\{v | v \in V, k = \mathrm{ind}(v)\}| \tag{4.24}$$

The probability for each $k \in B$ can be defined subsequently:

$$p^{BC}(k) = \frac{f^{BC}(k)}{\sum_{i=1}^{l} f^{BC}(i)} = \frac{f^{BC}(k)}{|V|} \tag{4.25}$$

The corresponding entropy and the relative entropy can be calculated as follows:

$$I_f^{BC}(G) = -\sum_{k=1}^{l} p^{BC}(k) * \ln\left(p^{BC}(k)\right) \tag{4.26}$$

$$\bar{I}_f^{BC}(G) = \frac{I(f^{BC})}{\ln l} \in [0, 1] \tag{4.27}$$

for $l > 1$.

4.4 Evaluation

4.4.1 Applying Information Functionals to Example Graphs

In this section we present the entropy values calculated using the information functionals f^V, f^J, f^D, f^{DS} and f^{BC}. To examine the behavior of the functionals, we

Graph	$\bar{I}_f^J$	$\bar{I}_f^V$	$\bar{I}_f^D$	$\bar{I}_f^{DS}$	$\bar{I}_f^{\mathrm{BC}}$	ρ	$\lvert V\rvert$
linear graph	0.904	0.376	0.99	1	0.843	7	8
star graph	0.863	0.941	0.991	0.591	0	2	8
CG	0	1	1	0	1	1	8
tree graph	0.92	0.515	0.99	0.967	0.947	5	8
circle graph	0.975	1	1	0	1	4	8

Table 4.2: Example Graphs. The parameters used to calculate $\bar{I}_f^V$ are: $\alpha = 0.5$, $c_1 = \rho$, $c_2 = \rho - 1, ..., c_\rho = 1$. The $\bar{I}_f^{\mathrm{BC}}$ is calculated on the distribution of BCs in the graph in analogy to $\bar{I}_f^{\mathrm{DS}}$.

selected some characteristic graphs of the same cardinality but differing in structure. These are a *linear graph* (a), a *star graph* (b), a *tree graph* (c), a *complete graph* (CG) (d), and a *circular graph* (e) (see Fig. 4.2). In addition, we calculate the entropy based on betweenness centralities of the graphs to compare the results with the outcomes produced by the functionals.

Table 4.2 lists the relative entropy values of these graphs. The f^D functional does not discriminate between linear and tree graphs (0.99), nor between CG and circle graphs (1.0). The star graph is slightly different but almost undistinguishable from the tree and linear graphs (0.991 vs. 0.99). The functionals f^V, f^{DS}, and f^J assign the lowest entropy (0.) to the CGs. The circle graph has the highest entropy for all functionals except for f^{DS} that has the opposite value (0 vs. 1). However, the f^J rates the circle graph with a value (0.975) below 1 . Increasing the entropy, the f^J, f^{DS} and $\bar{I}(BC)$ give the star graph the lowest entropy value after the CG. Here again, the f^J assigns a value above zero (0.863) to CG in contrast to the other two functionals. For f^V, the linear graph has the lowest entropy followed by the tree and the star graph. The CGs have always the entropy of 1 irrespective of the parameters used. This is a convenient property of f^V that allows to immediately filter out the CGs of an arbitrary size.

f^{DS} assigns the same entropy of zero to CG and to circle graphs, indicating that these different types of graph are not distinguished by the distribution of distance sums. Other example graphs are ranked by f^{DS} similar to the ranking of f^J, hence, with an increase of entropy we get: $\bar{I}(\text{star graphs}) < \bar{I}(\text{tree graphs}) < \bar{I}(\text{linear graphs})$.

According to these preliminary observations, the f^J functional behaves similar to the f^{BC} but has a higher discriminative potential. The f^D shows a poor discriminative ability on the graphs studied here. The f^{DS} performs better than f^D but it does not distinguish the CG from circular graphs. The f^V functional seems to weight the graphs differently, resulting in values different from the other functionals, especially

graph	(ρ)	$(\frac{\rho}{2})$	$(-)$	$(+)$
star graph	0.93	0.8	0.99	0.99
circle graph	1	1	1	1
CG	1	1	1	1
linear graph	0.33	1	0.99	0.99
tree graph	0.5	0.97	0.99	0.99

Table 4.3: Relative Entropy values for the Example Graphs using various parameter combinations. (ρ): $c_1 = 0, ..., c_{\rho-1} = 0, c_\rho = 1$, $(\frac{\rho}{2})$: $c_1 = 0, ..., c_{[\frac{\rho}{2}]} = 1, ..., c_\rho = 0$, $(+)$: $c_1 = 1, ..., c_{\rho-1} = \rho - 1, c_\rho = \rho$ and $(-)$: $c_1 = \rho, c_2 = \rho - 1, ..., c_\rho = 1$.

for star, linear and tree graphs. Obviously, the functionals reflect different topological properties of the graphs. In the next sections we will see how these functionals behave when applied to more complex networks.

4.4.2 Parameter Study for f^V

In this section, we look more closely at the functional f^V experimenting with parameters. First, we discarded the parameter α, as suggested in Remark IV.7, considering only the sum of cardinalities of j-spheres with corresponding coefficients. The resulting formula to compute the functional f^V was chosen as follows:

$$f^V(v_i) := (\bar{c}, \bar{S}(v_i)) = \sum_{j=1}^{\rho} c_j |S_j(v_i, G)| \tag{4.28}$$

We compared the results for different sets of parameters:

1. (ρ): $c_1 = 0, ..., c_{\rho-1} = 0, c_\rho = 1$
2. $(\frac{\rho}{2})$: $c_1 = 0, ..., c_{[\frac{\rho}{2}]} = 1, ..., c_\rho = 0$, for $\rho > 1$ else $I = 1$
3. $(+)$: $c_1 = 1, ..., c_{\rho-1} = \rho - 1, c_\rho = \rho$
4. $(-)$: $c_1 = \rho, c_2 = \rho - 1, ..., c_\rho = 1$

In the first case, the j-sphere with $j = \rho =$ diameter was weighted by 1 and all other spheres by 0. In the second case, the "middle" j-sphere (i.e., $j = \frac{\rho}{2}$) was weighted by 1 and the other j-spheres by 0. The last two alternatives weight the j-spheres by values from 1 to ρ increasing (or decreasing) with j.

Table 4.3 shows the results computed for the example graphs (shown in Fig. 4.2). We see from the table that star, circle and CGs have the same values for all four parameter combinations. Linear and tree graphs however, show a considerable difference for (ρ) on the one hand, and the other parameter combinations, on the other. That is, the same functional produces very diversified entropy values for the same

Graph	(ρ)	STD	$(\frac{\rho}{2})$	STD	$\lvert V\rvert$
German	0.46	-	0.99	-	195
English	0.53	-	0.99	-	163
RMDN_{LCC}	0.8	-	0.77	-	136
ER	0.882	0.253	0.989	0.003	195
ER	0.519	0.273	0.988	0.004	163
BA	0.600	0.078	0.958	0.002	195
WS	0.475	0.188	0.992	0.002	195

Table 4.4: Relative Entropy values using two parameter combinations (ρ): $c_1 = 0, ..., c_{\rho-1} = 0, c_\rho = 1$ and $(\frac{\rho}{2})$: $c_1 = 0, ..., c_{\frac{\rho}{2}} = 1, ..., c_\rho = 0$. ER, BA and WS graphs are presented in terms of average values and corresponding standard deviations (STD).

graph (e.g., 0.33 vs. 1 for linear graphs) when we vary the parameter c. Nevertheless, (ρ) and to some extent $(\frac{\rho}{2})$ distinguish different types of graphs from each other, which was not the case for the other two combinations.

4.5 Results

We use the parameters (ρ) and $(\frac{\rho}{2})$[5] to compute the entropy for English, German and Random-Word MDNs (RMDN), as well as for random graphs like *Erdős and Rényi* graphs (ER) and scale-free graphs (BA, WS).[6] The RMDN contains disconnected parts (see Fig. 4.7), thus, we calculate the entropy for the whole network as well as for the largest connected component (RMDN_{LCC}). Table 4.4 lists the resulting values calculated by means of f^V.

At first glance, the first parameter set (ρ) seems to produce more realistic results assigning lower values to German and English than to RMDN_{LCC}, and distinguishing well between the single graphs. However, we observe high fluctuations between single graphs of the same type as becomes evident from the high standard deviations (STD) of about 0.2. Figure 4.3 illustrates how the entropy values vary for ER graphs of 195 vertices compared to the values of f^{DS}, f^J and $f^V_{(\frac{\rho}{2})}$.

Furthermore, we look for the possibility to discriminate between the various types of networks by means of the information functionals discussed so far. We apply *Quantitative Network Analysis* (QNA) from *Mehler* (2008a, 2009) in order to learn classes of morphological networks by virtue of their structure, while disregarding content units (i.e., names of vertices). QNA basically integrates vector representations of

[5]We selected these combinations since they performed best in the parameter study shown in Table 4.3.

[6]We generate 10 graphs of each kind of random network (i.e., 10 graphs for ER, 10 for BA, etc.) and compare the averaged entropy values.

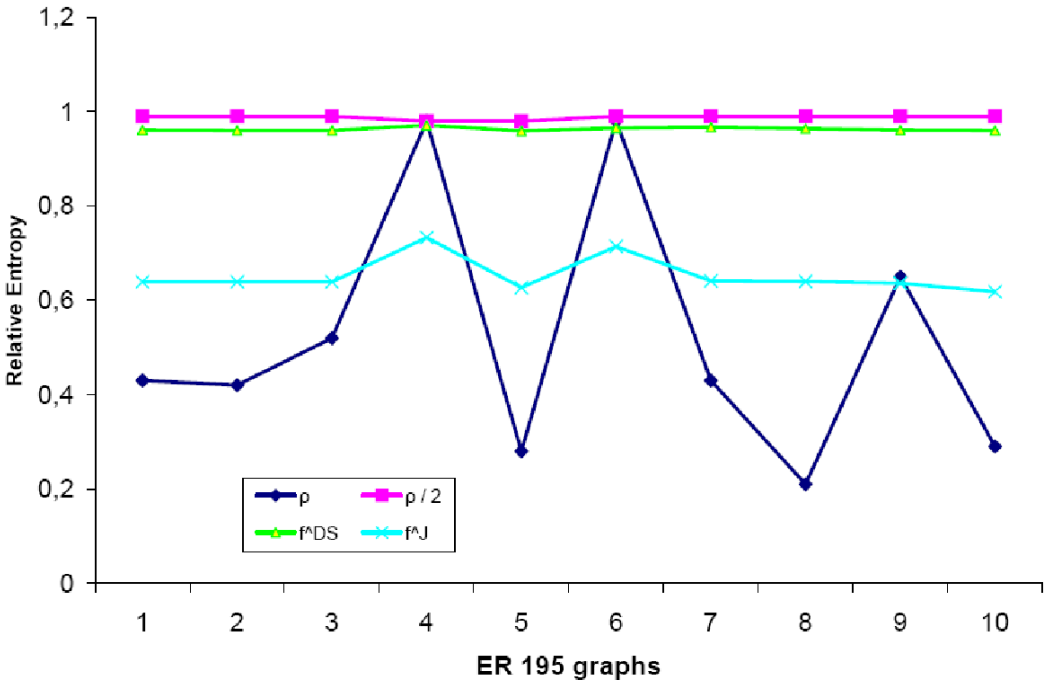

Figure 4.3: Comparison of relative entropy values (f^{DS}, f^{J}, $f^{V}_{(\rho)}$ and $f^{V}_{(\frac{\rho}{2})}$) computed for 10 randomly generated ER 195 graphs. Sets of parameters used for $f^{V}_{(\rho)}$ are: $c_1 = 0, ..., c_{\rho-1} = 0, c_\rho = 1$ and for $f^{V}_{(\frac{\rho}{2})}$: $c_1 = 0, ..., c_{\frac{\rho}{2}} = 1, ..., c_\rho = 0$.

complex networks with hierarchical cluster analysis. The cluster analysis is complemented by a subsequent partitioning, in which the number of classes is determined in advance (*Mehler*, 2008a). In this sense, QNA is semi-supervised (*Mehler et al.*, 2010a). The basic idea of QNA is to provide highly condensed numerical representations of networks that nevertheless capture their structural characteristics so that they can be automatically classified.

In our framework, QNA works as follows: given a vector representation of each graph (with dimensions representing entropy values based on 6 information functionals: f^{J}, f^{D}, $f^{V}_{(\rho)}$, $f^{V}_{(\frac{\rho}{2})}$, f^{DS} and f^{BC}) hierarchical clustering is applied. The algorithm examines several linkage methods (complete, single, average, weighted, centroid, median, ward) and distance metrics (mahalanobis, correlation) to find the best way to differentiate the data (*Mehler*, 2008a).

We use *F-Measure* statistics[7] to evaluate the classification. For the known partition of networks $\mathbb{L}$ and the partition found by the clustering algorithm $\mathbb{P}$ the F-measure is computed as follows:

$$\text{F-measure}(\mathbb{P}, \mathbb{L}) = \sum_{L \in \mathbb{L}} \frac{2 * \text{Recall}(P, L) * \text{Precision}(P, L)}{\text{Recall}(P, L) + \text{Precision}(P, L)} \in [0, 1] \tag{4.29}$$

For $P \in \mathbb{P}$ being the number of networks classified to a group, and $L \in \mathbb{L}$ the real

[7] See *Hotho et al.* (2005) for details.

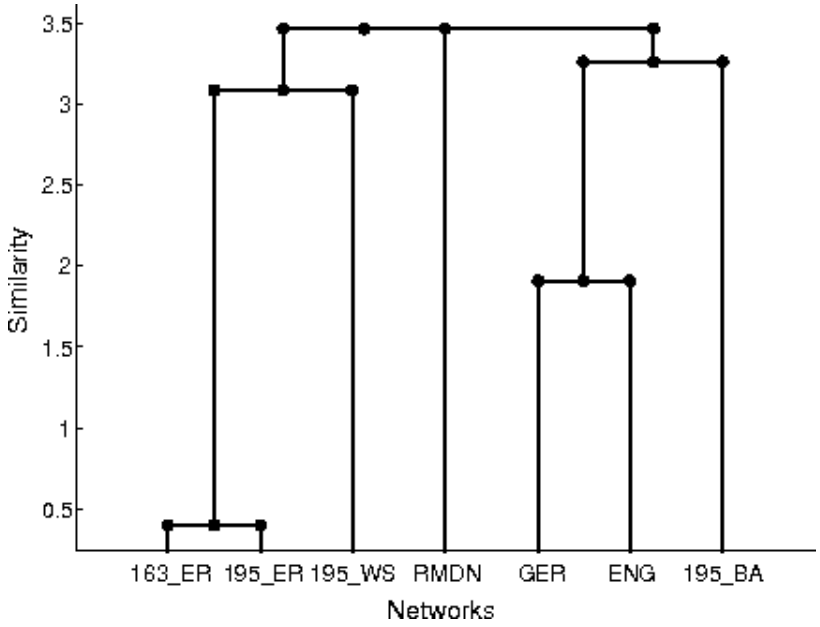

Figure 4.4: Clustering of graphs as feature vectors of 6 entropy values.

F-measure	random baseline	groups	distance	linkage	best features
1.0	.65848	5	mahalanobis	ward	$\bar{I}^J, f^D, f^{DS}, f^V_{(\frac{p}{2})}$

Table 4.5: Classification into 5 groups: (1) German and English, (2) ER, (3) RMDN, (4) WS, (5) BA. This is only one possible feature combination achieving an F-measure of 1. Running the genetic search for best feature 20 times, we obtained 11 different feature combinations responsible for the perfect classification of 1 in total. These combinations are shown in Table 4.6.

number of networks belonging to this group, $Precision = \frac{\#\{P \cap L\}}{\#\{P\}} \in [0,1]$ is the rate of correctly classified networks with respect to all networks classified to a group. $Recall = \frac{\#\{P \cap L\}}{\#\{L\}} \in [0,1]$ is the rate of correctly classified networks according to the total number of networks belonging to the group. The F-measure ranges from 0 to 1. A value close to 1 indicates that networks were classified correctly with respect to their group membership, and a value nearby 0 indicates that the classification have failed.

All entropy values used for classification are presented in Table 4.7.

4.6 Discussion

Figure 4.4 and Table 4.5 present the results of classifying the graphs by means of six functionals. All types of networks could be separated perfectly (F-measure of 1). German and English are clearly close to each other and can be distinguished by means of these measures. Concerning random graphs, the WS network is the most

similar one to English and German.

feature	combinations											sum
$\bar{I}_f^J$	✓	✓		✓				✓				4
$\bar{I}_{f_{(\rho)}}^V$							✓		✓	✓		3
$\bar{I}_f^D$	✓		✓		✓					✓	✓	5
$\bar{I}_f^{DS}$	✓	✓	✓	✓		✓	✓				✓	7
$\bar{I}_f^{\mathrm{BC}}$		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	10
$\bar{I}_{f_{(\frac{\rho}{2})}}^V$	✓		✓	✓	✓	✓	✓	✓	✓			8
sum features	4	3	4	4	3	3	4	3	3	3	3	

Table 4.6: Best feature combinations of QNA resulting in an F-measure of 1.0 found by the genetic search (which we ran 20 times).

Furthermore, we performed a genetic feature selection study filtering out the redundant features that do not improve the result of classification (Table 4.6). According to this study, the three best functionals are $\bar{I}_{f_{(\frac{\rho}{2})}}^V$, f^{DS} and f^{BC}, which are all based on distributions of different topological properties of graphs. f^{BC} is the most frequently selected feature, however, we ran the experiment only 20 times. While there are many different feature combinations already, presumably we will obtain even more if we continue our experiments. Furthermore, only four features are required to differentiate the networks perfectly. This is not surprising since the groups of networks are small (see Chapter VI for classification experiments dealing with larger sets of languages).

When we look at the relative entropy values more closely (Table 4.7), we see that English has a slightly higher entropy than German according to f^J, $f_{(\rho)}^V$, f^{DS} and f^{BC}. This result is in accordance with what we would expect comparing the use of derivational morphology in German and English. That is, German has a higher predictability of PoS by stems and suffixes than English. At the same time, most functionals assign lower entropy values to RMDN_{LCC} than to English and German, except for f^D and $f_{(\rho)}^V$; although, in the case of f^D the difference is very small. It seems more plausible to expect natural language networks to have lower entropies than random ones, like those obtained from $f_{(\rho)}^V$. However, the other functionals may take different properties of networks into account like, for example, the centrality of the graph. Regarding centrality, the RMDN is more centralized than the other two, which is visible in Figure 4.7. The values of f^{BC} confirm this fact assigning to RMDNs lower entropy values.

The functional f^D produces almost equal values for all graphs at the first and second decimal point (0.99). The functional f^{DS}, which is based on distributions of distance sums produces much better results.

graph	$\bar{I}_f^J$	$\bar{I}_{f_{(\rho)}}^V$	$\bar{I}_{f_{(\frac{\rho}{2})}}^V$	$\bar{I}_f^D$	$\bar{I}_f^{DS}$	$\bar{I}_f^{\mathrm{BC}}$	ρ	$\lvert V\rvert$
German	0.724	0.469	0.99	0.998	0.927	0.74	7	195
English	0.767	0.577	0.99	0.998	0.952	0.775	7	163
RMDN	0.71	-	-	0.997	0.68	0.398	7	136
RMDN$_{LCC}$	0.715	0.826	0.844	0.999	0.625	0.393	7	111
ER	0.666	0.882	0.989	0.999	0.960	0.968	4	195
ER	0.652	0.519	0.988	0.999	0.963	0.952	5	163
BA	0.714	0.600	0.958	0.998	0.977	0.744	6	195
WS	0.753	0.475	0.992	0.999	0.973	0.387	8	195

Table 4.7: Entropy measured using f^J, $f^V_{(\rho)}$, $f^V_{(\frac{\rho}{2})}$, f^D, f^{DS} and f^{BC}.

4.7 Summary

In summary, we were able to distinguish between language networks and the random ones by means of their entropy. Language networks differ much from the ER networks and from RMDNs, but are closer to WS networks, according to their entropy values. Furthermore, the MDNs from natural languages can be distinguished from random ones by means of their topological characteristics. This finding encourages the use of network approaches in typological studies. That is, constructing an MDN of a language allows us to examine its morphological properties that can be learned from the network topology.

In addition, we studied some information functionals each of them seems to highlight a different aspect of the graph, either the distribution of j-spheres, of the shortest distances between vertices, or of the distance sums in the graph. The entropy based on these functionals allows a perfect distinction of natural language networks from RMDN as well as from random graphs (ER, BA, WS). Information functionals based on distributions of topological properties turned out to be better discriminators than those that are based on properties of single vertices (e.g., f^D vs. f^{DS}).

MDNs are an example of morphological networks that capture only one aspect of morphology, namely derivation by means of suffixes. However, these networks contain information about the organization principles of languages that become apparent from their topology. This was demonstrated by our approach. Extensions of the network model including other kinds of morphemes (identified e.g., by means of a morpheme-segmentation algorithm) should complete the picture. Future work aims to study more sophisticated network models of morphology and their application in typological research.

Acknowledgement

We would like to express our gratitude to Alexander Mehler and Kirill Medvedev for fruitful discussions and comments. Our special thanks goes to Matthias Dehmer whose useful hints and recommendations helped to improve this paper.

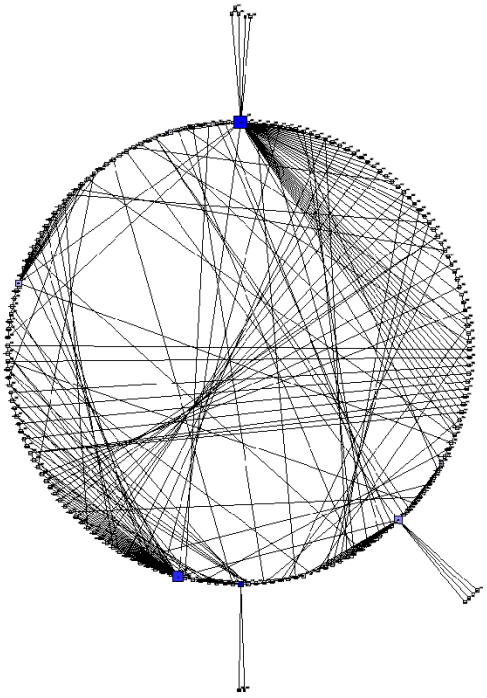

Figure 4.5: German MDN. Visualization of the Betweenness Centralities. The three most central vertices: *Noun*, *-en* suffix, *Adjective*.

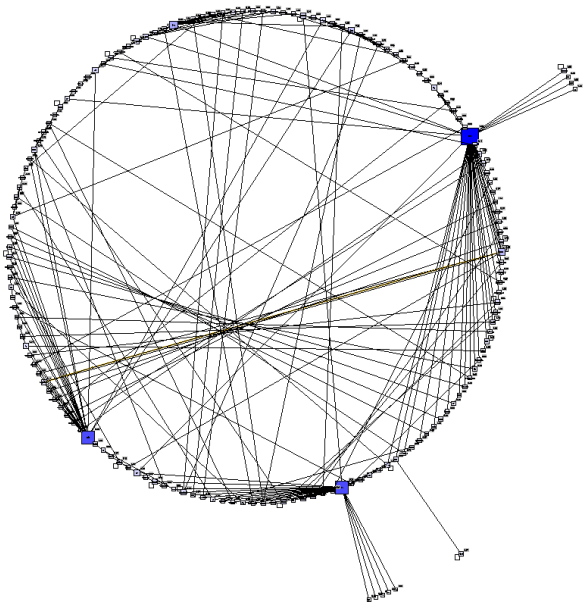

Figure 4.6: English MDN. Visualization of the Betweenness Centralities. The three most central vertices: *Noun*, *Adjective*, *Verb*.

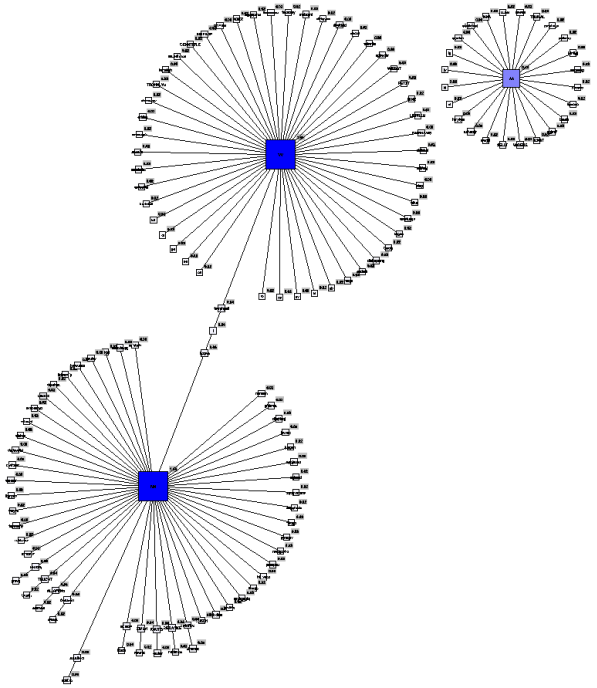

Figure 4.7: Random MDN. Visualization of the Betweenness Centralities. The three most central vertices: *Noun*, *Verb*, *Adjective*.

CHAPTER V

Phonological Networks

5.1 Introduction

In this chapter we compare the phoneme inventories of languages (from the UPSID[1]), and try to reconstruct genetic relationships by means of them. The main assumption behind this approach is that, after they have split apart, the relationship between two languages continues to exist in their sound systems. If these traces are systematic they might correspond to genetic distances of languages. But this is not necessarily the case. Sound systems might undergo various processes of change that are not necessarily consistent with the change rate of languages within a language family. Comparing languages by the amount of common phonemes we try to answer the following questions:

- Does the phonetic space two languages share tell us something about their genealogical distance?
- Do phonological inventories allow us to distinguish between different language families?
- Is it possible to reconstruct the distances between languages of a single language family by means of phonological inventories?

The method presented here allows to compute phonological similarity and achieves good results in classifying languages in genealogical groups. Some languages, however, are misclassified. By inspecting the outliers, we could find an isolated language, Ainu, that was classified to the Papuan language family by our method with a high degree of similarity. This could be a hint to linguists to rethink the classification of Ainu. This example shows that the presented method can serve as an additional means to test the genealogical relationship of languages or to verify the formation of a language family.

[1] i.e., the UCLA Phonological Segment Inventory Database (UPSID) (*Maddieson*, 1984; *Maddieson and Precoda*, 1989) which is discussed in the following.

5.2 Related Work

Tambovtsev (2007) defines the typological distance between languages in terms of *phonostatistics* - or frequencies of occurrence of phonemes. In analogy to QNA (Chapters IV, VI), language distance is computed based on quantitative profiles consisting of phonological features computed for each language. *Tambovtsev* (2007) concentrates on consonants, since "consonants bear the semantic load in the word, not vowels" *Tambovtsev* (2007, 77). However, his method can be extended to include vowels too. To create the quantitative profiles of a language, *Tambovtsev* subsumes the consonants into 8 articulatory groups.[2] Rrather than counting the occurrence of a single consonant, he counts the frequencies of all phonemes representing each of the 8 groups. Moreover, he calculates the proportions of consonants, of vowels and the ratio of consonants to vowels as additional features. To calculate the distance between two languages, he applies the Euclidean distance between the single features of the quantitative profile:

$$D(l_1, l_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (...)^2 + (N_1 - N_2)^2} \tag{5.1}$$

D represents the Euclidean distance between L1 and L2 and N features.[3] x_1, x_2 are the frequencies of occurrence of, for example, labial consonants in L1 and L2, y_1, y_2 are the frequencies of occurrence of forelingual consonants in L1 and L2, etc.

Tambovtsev (2007) examines the closeness of Latin to some Romance languages by comparing their distances D. Our approach is similar to Tambovtsev's, we also operate with quantitative profiles but we use different kinds of features.

Comparing languages based on phonetics, we define the distance between them as the amount of common phonemes the two languages share. We use the UCLA Phonological Segment Inventory Database (UPSID) (*Maddieson*, 1984; *Maddieson and Precoda*, 1989) to compare the phonetic inventories of languages. The UPSID is a data base containing information about phonetic inventories of languages and language families the languages belong to. *Maddieson* (1984) presents a detailed description of sound patterns based on the UPSID and points to the directions of research that can be pursued using the database. Thus, the UPSID represents a potentially useful resource for typological research.

However, *Simpson* (1999) cautions against using phonological databases like UPSID for typological research. He claims that the selection of phons representing the group of allophones is done arbitrarily. There are many different criteria to select the phon representing a group, like for example, a phon with the most articulatory centricity, the highest frequency, the least affectation by the context, occurring in isolation, etc. *Simpson* admits that the UPSID is constructed pursuing the same criteria for the selection of the representative-phon. Although the creators of the UPSID do not explicitly state which criterium they selected, *Simpson* (1999, 350) assumes that "the UPSID phoneme evidently bears a strong resemblance to a Jonesian or American structuralist family or group of allophones."

[2]The groups are: labial, forelingual (front), palatal (mediolingual), guttural (back), sonorant, occlusive non-sonorant, fricative non-sonorant and voiced non-sonorant.

[3]L1 means the first and L2 the second language a person acquires.

Simpson mainly criticizes that constructing the UPSID the selected representative-phon stands for the whole group, while other allophones are not considered.

> "The allophone no longer represents the phoneme, it *replaces* it; the phoneme and its characteristic allophone become one and the same thing." (*Simpson*, 1999, 350)

As a result, the phoneme inventories of the UPSID represent a sort of abstraction of the total phonetic space used by languages. Of course, studies of the role of various allophones within a phoneme, as well as, the interdependence between allophones of different groups are not feasible by means of such databases. And conclusions made using such databases should be aware of this reduction. For the purpose of our work, however, such phoneme-representatives might suffice since we do not aim to analyze the inner phoneme variation but rather at inter language comparisons by means of phoneme inventories.

Mukherjee et al. (2009) present an approach to model the distribution of consonants among languages in a bi-partite network. More specifically, they introduce two types of networks - the Phonetic Language Network (PlaNet) and the Phonetic Network (PhoNet). Both networks are induced from the UPSID comprising 317 languages and 541 consonants. PlaNet consists of two types of vertices - languages and phonemes - that become connected if a phoneme occurs in a language. PhoNet is a projection of PlaNet in which two phonemes are linked when they co-occur in a language. *Mukherjee et al.* (2009) calculate the degree distributions of both networks; a β-distribution in the case of PlaNet and a power-law distribution between two cut-off points of PhoNet. Additionally, they compute the weighted clustering coefficient (*Barrat et al.*, 2008) on PhoNet that exhibits high clustering. Finally, they present a synthesis model that generates networks of the kind observed empirically (i.e., PhoNet and PlaNet). The analysis of community structure in PhoNet have shown that consonants which are predominant across languages of the world exhibit strong co-occurrence patterns. They explain this emergence of communities by the force to use the same features. This leads to a small number of distinctive features and a larger amount of possibilities to combine them (*Choudhury and Mukherjee*, 2009).

This approach, though very promising, does not attempt to compare languages by means of their phonetic inventories (cf. *De Boer* (2001); *Jäger* (2006)). In this chapter we will fill this gap. Our approach relies on the work of *Mukherjee et al.* (2009) and is related to the work of *Kapatsinski* (2008), who uses the UPSID to compare the phoneme inventories of languages to each other.

5.3 Approach

To answer the questions posed in the introduction of this chapter, we use the *Quantitative Structure Analysis* (QSA) as the general form of QNA (described in Chapters IV and VI).[4] According to it, each language is represented by a feature vector

[4]See also *Mehler et al.* (2007) for details on QSA.

whose dimensions capture the phoneme space of the UPSID. The feature values of each phoneme-feature are binary, indicating whether the particular phoneme is contained in a language or not. We apply cluster analysis (*Mehler*, 2008a) to compare languages by means of these feature vectors. To answer the questions from the introduction we perform different kinds of experiments:

1. **Language Family Relationship**.
    a. In the first experiment we select 3 language families from the UPSID, 11 languages in total, and perform the cluster analysis. The algorithm tries to assign languages to the four groups according to the similarity of the feature vectors. The results are given in terms of *F-measure* statistics (harmonic mean between precision and recall, see Chapter IV). The corresponding random baselines are calculated.
    b. In the second experiment, we increase the number of language families to 5, select the classes ($\sim$ 20 languages each), and repeat the procedure of 1a.
    c. In the third experiment, we increase the size of the family taking 3 families of $\sim$ 100 languages each, and repeat the procedure of 1a.
2. **Inner Group Similarity**. In this experiment, we compare the similarities of languages within a group by means of a dendrogram. The dendrograms created by means of phonological similarity are compared to established language tree classifications.
3. In this series of experiments we look at the similarities between pairs of languages and compare the results to the similarity measuremnts obtained by the approach proposed in *Tambovtsev* (2007).

5.4 Experiments

5.4.1 Measuring the inner-language-family distance

In these series of experiments we try to falsify the null hypothesis that phoneme inventories have no correlation with language family membership.

5.4.1.1 Experiment 1a: 5 Families and 118 languages in total.

The language families classified in this experiment are presented in Table 5.1. To account for an approximately equal probability of each language family, we have selected five families of roughly equal size of about 20 languages. The goal is to classify 118 languages into 5 groups, which is not an easy task for the algorithm due to the large number of objects and features. The results are shown in Table 5.2.

We ran the genetic search for best features 10 times. The average F-measure value over 10 trials is .6496. Considering all features (i.e., 500) results in an F-measure of .54053. This value is lower than the F-measure obtained by reducing the feature space but still above the random baseline of .30. The baselines are clearly surpassed in all

language family	languages	sub-families
Afro-Asiatic	26	6
Australian	25	9
Indo-European	23	12
Nilo-Saharan	23	8
Sino-Tibetan	21	8

Table 5.1: Language families selected by the number of languages ~ 20. The number of sub-families indicates the heterogeneity of a language family.

procedure	F-score	method	class
`QSA[correlation,hierarchical,complete]`	.68861	465/500	5
`QSA[correlation,hierarchical,complete]`	.68541	464/500	5
`QSA[correlation,hierarchical,weighted]`	.68376	345/500	5
`QSA[euclidean,hierarchical,complete]`	.54053	500/500	5
`AVG` (over non-random approaches)	.6496		
`random baseline I`	.30222	equi-partition	
`random baseline II`	.30003	known partition	

Table 5.2: Experiment 1a: Language family relationships between languages of 5 families is tested using all features (the total phonetic space of 500), and features chosen by a genetic feature selection algorithm.

cases, so we are able to clearly distinguish five language families of approximately equal size by means of their phoneme inventories.

5.4.1.2 Experiment 1b: 3 Families and 177 languages in total.

In this experiment we take the three largest families from the UPSID - the Niger-Kordofanian (55 languages), the North American (56 languages) and the South American (66 languages). We increase the number of languages within the family and simultaneously decrease the number of language families (from five to three). The

language family	languages	sub-families
Niger-Kordofanian	55	14
North American	56	8
South American	66	9

Table 5.3: Language families with the highest number of languages in the UPSID.

F-measure rises to .82485 for a reduced number of classes, although more languages are to be classified. The results still surpass the random baselines, thus, the genealogical classification succeeds.

procedure	F-measure	features	class
QSA[correlation,hierarchical,complete]	.82485	474/500	3
QSA[correlation,hierarchical,complete]	.81606	461/500	3
QSA[correlation,hierarchical,complete]	.80077	495/500	3
QSA[correlation,hierarchical,complete]	.66533	500/500	3
AVG (over non-random approaches)	.7768		
random baseline I	.38848	equi-partition	
random baseline II	.38764	known partition	

Table 5.4: Experiment 1b: Language Family Relationship of 3 language families.

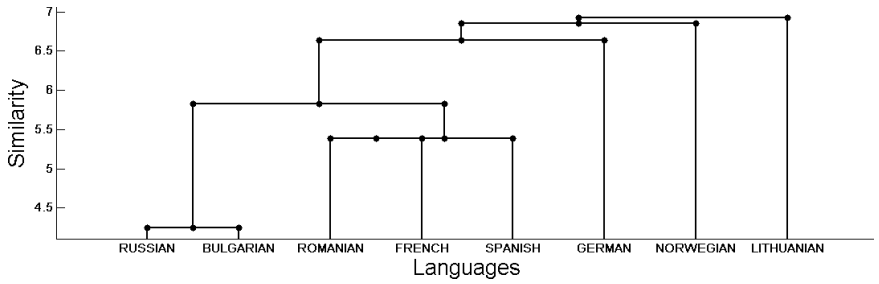

Figure 5.1: Clustering of 8 Indo-European languages from 5 sub-groups: Slavic, Romance, West-Germanic, North-Germanic and Baltic.

5.4.2 Measuring the distances between sub-groups of a single language family

procedure	F-measure	languages	class	random I	random II
QSA[eucledian,single]	1	5	3	.63075	.65118
QSA[eucledian,single]	1	8	5	.62245	.65686
QSA[eucledian,single]	.78333	14	6	.55173	.56323
QSA[eucledian,ward]	.69004	23	12	.49736	.50476

Table 5.5: Experiment 2: Similarity within the Indo-European family. All classifications surpass the random baselines, which are the average values over 100 random iterations. Random I assumes an equi-partition of languages into classes, Random II takes the actual cardinality of the entire groups.

In this section, we concentrate on the Indo-European language family and try to cluster its sub-groups. The results show that five language subfamilies (Baltic, Romance, North-Germanic, West-Germanic and Slavic) can be separated distinctly from each other by means of their phoneme inventories (Table 5.5). The F-measure decreases when we consider more subfamilies, although it does not fall below the random baseline. This result points to a high variability of languages within the Indo-European family, which makes implicit comparison of Indo-European with other language families problematic.

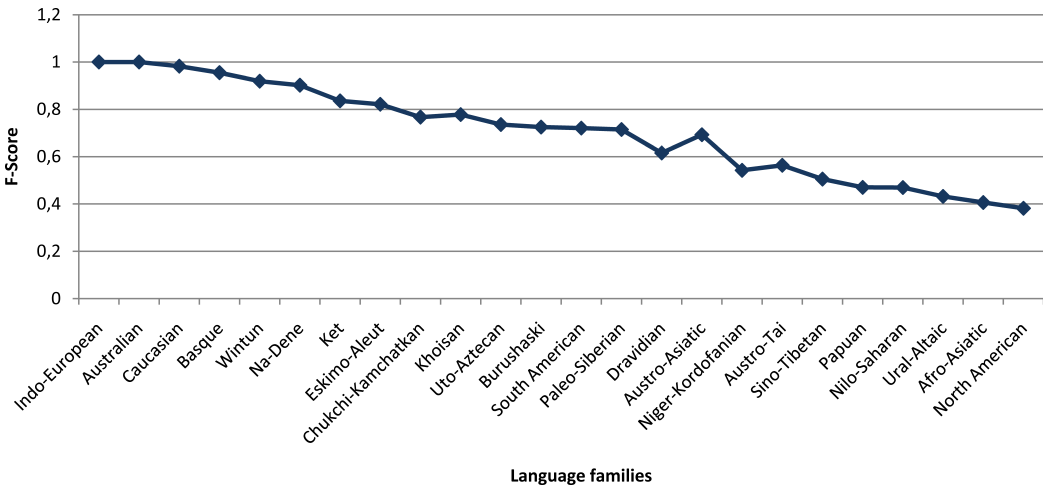

Figure 5.2: Language Families ranked according to the greatest dissimilarity. Starting at the seed containing Indo-European, language families that are best distinguished from the seed are incrementally added to the seed. The mean F-score of all language families is: .7055 and the standard deviation: .1992. The F-score of the random baseline (known-partition) averaged over 1000 trials is .16028 and for the equi-partition .14837 respectively.

The dendrogram in Figure 5.1 shows the similarities between 5 sub-groups of Indo-European that are mostly plausible (cf. e.g., German and Norwegian connect together and are both representatives of two Germanic sub-groups, West- and North-Germanic).

5.4.3 Ranking of language families

To gain deeper insight into the overall role of language (sub-)families for the resulting classification we apply the *Iterative Categorisation Procedure* (ICP) proposed in *Pustylnikov and Mehler* (2007). Starting with an input seed set of language families (e.g., Indo-European in Figure 5.2), all resulting families are incrementally added to the seed and the F-score is calculated. The family with the best F-score, that is, the most dissimilar to the families in the seed is retained. This procedure is repeated until all language families are contained in the seed.

1. **Start** ($i = 1$)**:** We select a seed language family and assign the rank number 1 to it.

2. **Iteration** ($i > 1$)**:** In the i-th step we check the F-score while adding in turns each of the remaining B_i families to the new set. The family with the minimal F-score decrease (i.e., the greatest dissimilarity to the seed) is finally added to the result set, receiving the last rank position.

3. **Break off:** The rank ordering is complete IF:

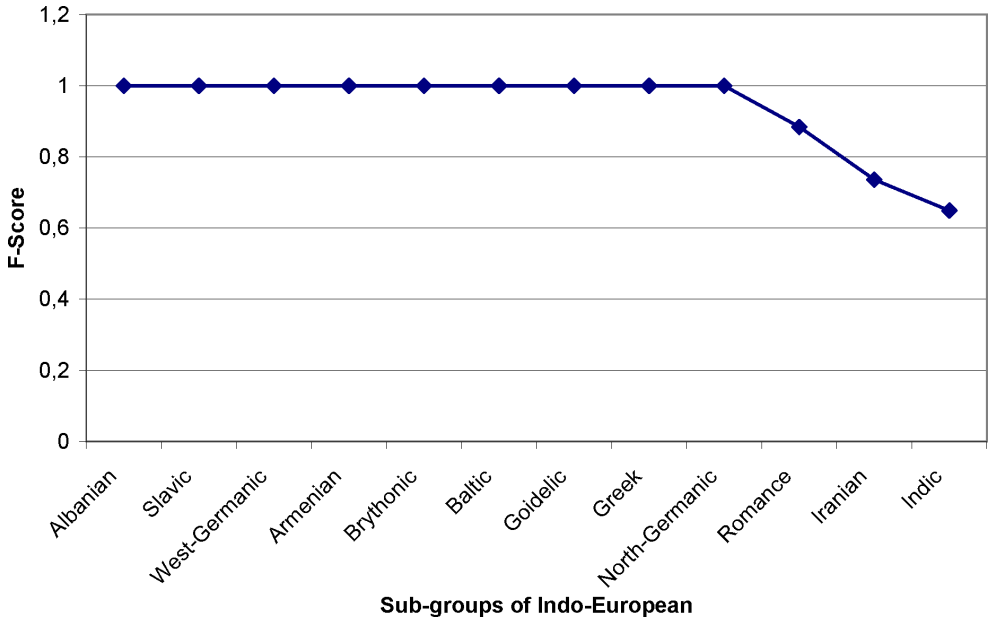

Figure 5.3: Sub-groups of Indo-European ranked according to the greatest dissimilarity. Starting from a seed containing Albanian, sub-groups that are best separated from the seed are incrementally added to seed. The mean and standard deviation of classifying Indo-European sub-groups are .93917 and .12118 respectively.

i) all families are ranked within the result set

ii) the predefined *cut-off* (e.g., $c = 0.4$) is reached

iii) the F-score of the result set in combination with each category of B lies under a specific baseline.

This kind of ranking is more informative than the F-score values alone. That is, the F-score adds global information on the overall separability of the families. The ICP, in contrast, provides additional information on the overall dissimilarity among single families. The results of ranking the language families available in UPSID are shown in Figure 5.2. Indo-European is most distant from Australian languages (F-score= 1), then, the F-score starts to drop. However, the mean F-score value for this classification is .7055 which clearly surpasses the baseline of .16028 (or .14837 respectively) (none of the language families worsens the classification to an F-score below the baseline). This result shows that notwithstanding the inner variability of single language families like Indo-European, a language family is still a class with its particular distinctive phonological characteristics. The changes languages undergo over time do not completely eliminate their family resemblance with respect to phoneme inventories.

The standard deviation of this classification experiment is .1992. There can be several reasons for this high standard deviation. One reason might be that the high similarity among single families prevents the algorithm from separating them clearly.

Another reason could be the high internal heterogeneity of some families, that is, languages within a family can be dissimilar to each other. Other reasons can be presumably attributed to the varying size of the groups as well as to the large number of objects that can both deteriorate the result of the classification.

In order to examine the separability of languages within a language family, we concentrated on the highly heterogenous Indo-European group consisting of 12 sub-groups. In this experiment, one group (i.e., the Albanian in this case) is selected as the seed (see Fig. 5.3), and other languages are incrementally tested for the best classification. The high mean F-score= .93917 indicates the overall high separability of language-groups within the IE-family. That means, IE languages differ that much from one another that allows for a good separability within this group.

Looking more closely at the tail of the curve in Figure 5.3, we see that the Iranian, Indic and Romance languages negatively influence the classification. We ran the ICP procedure again on a smaller set of categories, selecting the homogenous Slavic group as the seed. The resulting ranking is the following: Iranian (1.0) > Indic (.91751) > Romance (.71919).[5]

We selected each of the resulting IE sub-groups as a seed and Iranian, Indic and Romance as the additional languages and ran the procedure. The ranking did not differ much - Indic and Romance always occupied (sometimes interchangeably) the last ranks. According to this result, Indic and Romance do not seem to represent a homogeneous group in phonological terms, since these groups can not be clearly distinguished from the other IE sub-groups.

5.4.4 Measuring the similarity between sub-groups of different language families

In this section, we experiment with different sub-groups of the same or different language families and try to distinguish them by means of our approach. There are many different sub-groups that can be compared to each other according to phonological profiles. In the following, we will restrict our investigation to three following experiments.

5.4.4.1 Indic, Slavic, Greek and Romance

First, we examine the critical Romance group and plot it in comparison to Indic, Slavic and Greek according to the phonological distance. As plotted in Figure 5.4 the Slavic, Romance and Indic languages are correctly placed into different clusters. The Greek language, which represents the Greek family, exhibits the greatest similarity to Romance languages and is placed within the Romance cluster. Russian and Bulgarian constitute one Slavic cluster and the Indic languages are grouped close to each other but with a higher distance (indicated by the height of the bar). Nepali and Bengali

[5]The ranking can differ slightly depending on the initial seed, the average value, however, is the same for all combinations.

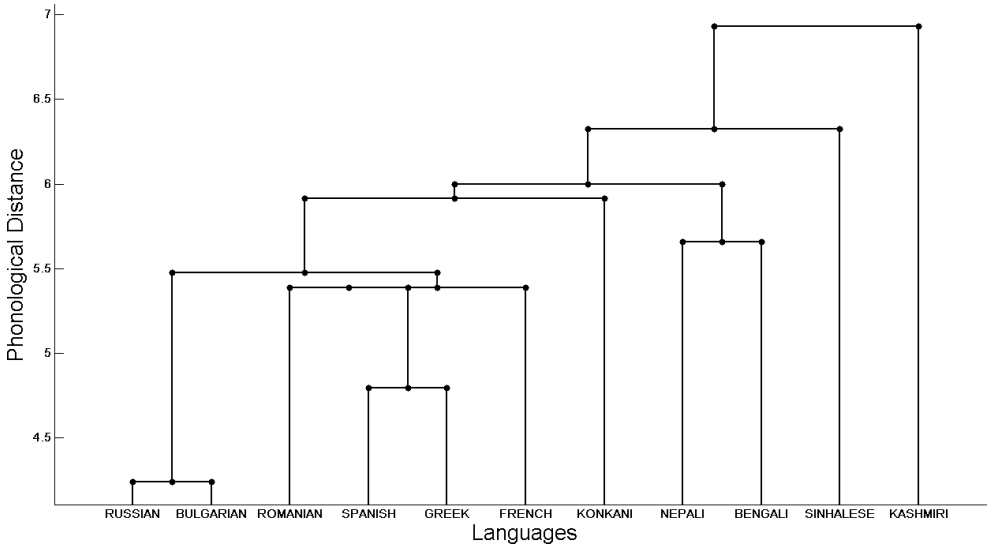

Figure 5.4: Indic, Slavic, Greek and Romance languages clustered according to their phonological distance.

form a cluster which corresponds also to the two languages' geographic location[6] (see Fig. 5.5).

5.4.4.2 West-Germanic, Slavic, Turkic and Romance

In this experiment the phonological distance of Turkic is compared to Slavic, Romance and West-Germanic (i.e., German) languages. Figure 5.6 indicates that these languages can be clearly divided into clusters with respect to their genealogical relationship. The areal relatedness of Turkic languages is also reflected by the clusters. Kirghiz and Uzbek, which form a further cluster, are also closely situated geographically, while more distant from Turkish and Chuvash, which are more distant from each other geographically, as well as in terms of clusters.

5.4.4.3 Reconstructing the phonological distances of (*Tambovtsev*, 2007)

In the last experiment we compare the results of distance measuring by means of D (Table 5.6) proposed in (*Tambovtsev*, 2007) to the (Euclidean) distances produced by comparing the phoneme inventories of the UPSID (Table 5.7). *Tambovtsev* compares Japanese to different languages and reports the distances organized increasing the distance D in Table 5.6.

At first glance, the UPSID-based method produces completely different results to the approach of *Tambovtsev*. However, the results do not differ as strongly as it may appear. *Tambovtsev* observed the similarity between Japanese and Altaic languages including "Turkic, Mongolian and Tungus-Manchurian" *Tambovtsev* (2007, 82). Our

[6]See *Kapatsinski* (2008) who also observed areal and genealogical similarity present in the UPSID data.

Figure 5.5: Geographical distribution of some Indic languages from WALS *Haspelmath et al.* (2005).

languages	**distance** D
Jap-Tur	9.02
Jap-Ket	9.52
Jap-Uzb	10.63
Jap-Hau	10.98
Jap-Geo	11.05
Jap-Rum	15.08
Jap-Ger	22.24

Table 5.6: Distances of Japanese to seven languages: Japanese (Jap), Uzbek (Uzb), Ket, Hausa (Hau), German (Ger) and Georgian (Geo) (*Tambovtsev*, 2007).

results also show this closeness - Japanese is less distant to Turkish and Uzbek than to German. When we include other Turkic languages (i.e., Azerbaijani, Kyrgyz) as well as an Iranian language (Kurdish) we see that the similarities are consistent (see Tab. 5.8). However, our results differ from the results of *Tambovtsev* in the cases of Georgian and Romanian.

According to *Tambovtsev*, Georgian is more similar to Japanese than in our case. However, Georgian is more distant from Turkic languages in both studies, so presumably this deviation is not significant. Romanian is one of the most distant languages according to *Tambovtsev*. In our study, in contrast, Japanese is most similar to Romanian. Here, presumably different typological accents are responsible for the bias. *Tambovtsev*'s method is presumably more precise here since it distinguishes between different consonantal groups, while it disregards vowels, as opposed to our method. Typologically, Romanian exhibits similarities to Turkish languages due to language contacts, thus, the smaller distance to Japanese, which is also close to Turkish, can be explained. These, however, are only speculations - more elaborate typological re-

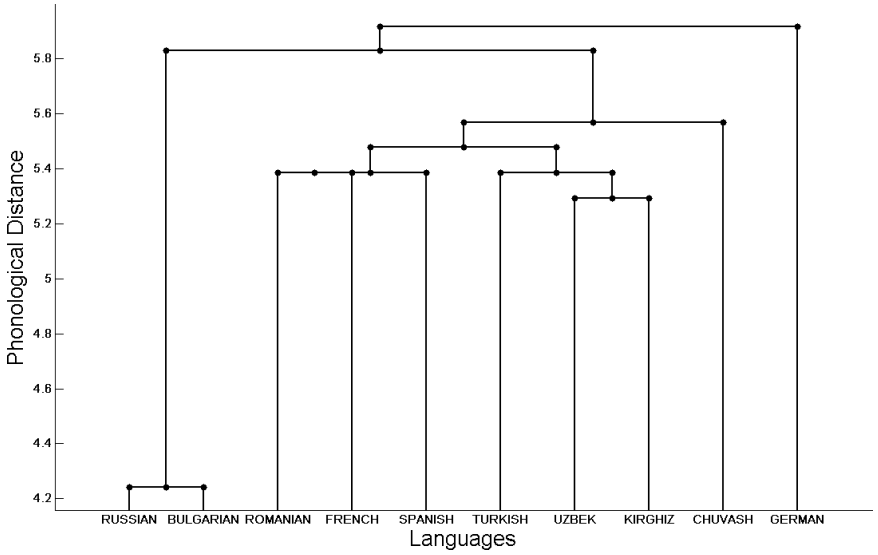

Figure 5.6: West-Germanic, Slavic, Turkic and Romance languages clustered according to their phonological distance.

search is needed to evaluate the validity of the quantitative results obtained in our study and in (*Tambovtsev*, 2007).

Here, we can confirm the general findings about the phonological closeness of Turkic languages and Japanese observed by *Tambovtsev*, and larger distances between Japanese and German.

5.5 Language Typology by means of LaPNet

Following the approach of *Mukherjee et al.* (2009) we use the UPSID to construct a language network. As discussed in the introduction to this chapter, *Mukherjee et al.* (2009) introduce two types of networks PhoNet and PlaNet for modeling the distribution of phonemes in languages. We extend the above approach by introducing the Language Phoneme Network (LaPNet), which is used to study language relationships based on phonetic similarity. Section 5.5.1 describes how the networks were obtained. In Section 5.5.5 we present a typology of languages based on the topology of LaPNets.

5.5.1 Network Definition

To construct language networks based on phonetic similarity, we start from the approach of *Kapatsinski* (2008). In the first step, we construct a language-phoneme coincidence matrix from the UPSID. According to (*Maddieson*, 1984, 196):

> "The languages [in the UPSID][7] are chosen to represent a properly structured quota sample of the genetic diversity of extant languages. One

[7]Commented by O.A.

Figure 5.7: Geographical distribution of Turkic languages from WALS *Haspelmath et al.* (2005).

languages	**distance** *UPSID*
Jap-Rum	4.6904
Jap-Uzb	5.099
Jap-Tur	5.3852
Jap-Ket	5.5678
Jap-Hau	6.3246
Jap-Ger	6.5574
Jap-Geo	6.6332

Table 5.7: Phonological distances computed based on the UPSID and Euclidean distance metric for languages: Japanese (Jap), Uzbek (Uzb), Ket, Hausa (Hau), German (Ger) and Georgian (Geo).

> and only one language is included from each cluster of related languages judged to be separated from its nearest relative to a degree similar to the separation of North and West Germanic (taken to be equivalent to about 1500 years of separate development)."

Thus, each language constitutes a representative of a particular genetic sub-group or family.

In contrast to *Kapatsinski* (2008), we utilize the total number of 919 phonemes present in the UPSID (cf. *Kapatsinski* (2008) omits 'dental', 'alveolar' and 'unspecified dental/alveolar' segments). The languages x phonemes (425 x 919) zero matrix is constructed. According to *Kapatsinski* (2008), if a phoneme Y is present in a language X, the cell in the X^{th} row of the Y^{th} column of the matrix is set to 1, otherwise its value remains zero. In more formal terms, for a set $\mathfrak{L}$ of 425 languages and $\mathfrak{P}$ of 919 phonemes a binary coincidence matrix $T = \{t_{ij}\}$ is constructed, where $i = \{1, 2, ..., \mathfrak{L}\}$ and $j = \{1, 2, ..., \mathfrak{P}\}$.

languages	**distance** *UPSID*
Jap-Rum	4.6904
Jap-Uzb	5.099
Jap-Tur	5.3852
Jap-Aze	5.3852
Jap-Kir	5.3852
Jap-Ket	5.5678
Jap-Kur	5.6569
Jap-Hau	6.3246
Jap-Ger	6.5574
Jap-Geo	6.6332

Table 5.8: Phonological distances computed based on the UPSID and Euclidean distance metric including Azerbaijani (Aze), Kyrgyz (Kir) and Kurdish (Kur).

In the second step, we calculate the number of shared phonemes between two languages l_i and l_j using T. We obtain:

$$K(l_i, l_j) = (\bar{t_i} * \bar{t_j}) \tag{5.2}$$

where $\bar{t_i}$ and $\bar{t_j}$ are the row vectors of the i-th and j-th rows in T, and $\bar{t_i} * \bar{t_j}$ is the scalar product $\sum_{k=1}^{|\mathfrak{P}|} t_{ik} * t_{jk}$.

Now, we define a similarity index s_{ij} between any l_i and l_j as follows:

$$s_{ij} = \frac{K(l_i, l_j)}{N_i} \in [0, 1] \tag{5.3}$$

where N_i corresponds to the number of 1-entries in $\bar{t_i}$. Note, that the matrix $S = \{s_{ij}\}$ of similarity values is asymmetric in general. The symmetry axiom $s_{ij} = s_{ji}$ does not necessarily hold, and in order to perform distance measuring geometric models, taking asymmetry into account should be considered (see Sec. 5.5.4 below).

Let $\theta \in [0, 1]$ be a similarity threshold. Let further M_i be the maximal value of the i-th row in S. Now, we are able to define the language-phoneme networks (LaPNet).

Definition 8. *We define $G(V, E)$ as a simple directed weighted graph (without graph-loops and multiple directed edges). Vertices of G $\{v_1, v_2, ..., v_L\} = V$ correspond to languages $\{l_1, l_2, ..., l_L\} = L$. A directed weighted edge $e_{ij} \in E$ from any $v_i \to v_j$ where $v_i \neq v_j$ is formed iff $s_{ij} > \theta \wedge s_{ij} = M_i$, where M_i is the maximal value of the i-th row of the matrix S.*

Remark V.1. The threshold θ allows to vary the degree of similarity between two languages, that is, if the similarity s_{ij} is below θ the language vertices v_i and v_j are not linked.

Remark V.2. The second condition $s_{i,j} = M_i$ guarantees that a vertex v_j which exhibits the greatest similarity to v_i is taken to form an edge with it.

We define the networks that way due to the following reasons:

- Decreasing (or increasing) the parameter θ (Definition 8 and Remark V.1) allows us to include more (or less) similar pairs of languages into the network.

- The second condition (Remark V.2) ensures that each language connects only to the most similar other.[8]

- If $i \rightarrow j \wedge j \rightarrow i$ for two languages i and j, we speak of a mutual similarity between them. This means, if two languages share an edge in both directions, we can speak of the highest similarity between them.

In fact, as we will see later from the resulting networks, many of these mutual similarities between languages also indicate their genetic relationship.

5.5.2 Constructing LaPNets using the similarity index s

This section summarizes the main steps performed to construct a series of LaPNets $G_\theta(V, E)$ depending on θ, which are in detail described in Section 5.5.1. Based on the data from the UPSID, we construct LaPNets $G_\theta(V, E)$ using the following procedure:

Require: matrix T
Ensure: LaPNets $G_\theta(V, E)$ for $\theta = \{0, 0.05, ..., 1\}$
1: Set $\theta = 0$
2: Calculate the asymmetric similarity matrix S, vector M of maximal values in each row in S
3: **while** $\theta \neq 1$ **do**
4: $E = \emptyset$
5: **for** $i = 1$ to $|L|$ **do**
6: **for** $j = 1$ to $|L|$ **do**
7: **if** $i \neq j \wedge s_{ij} > \theta \wedge s_{ij} == M_i$ **then**
8: add the edge e_{ij} to E
9: **end if**
10: **end for**
11: **end for**
12: print $G_\theta(V, E)$
13: $\theta = \theta + 0.05$
14: **end while**

The above procedure creates a series of LaPNets increasing the similarity threshold θ, which results in a smaller number of vertices in G. Thus, the higher the threshold θ, the less languages are included into the network.

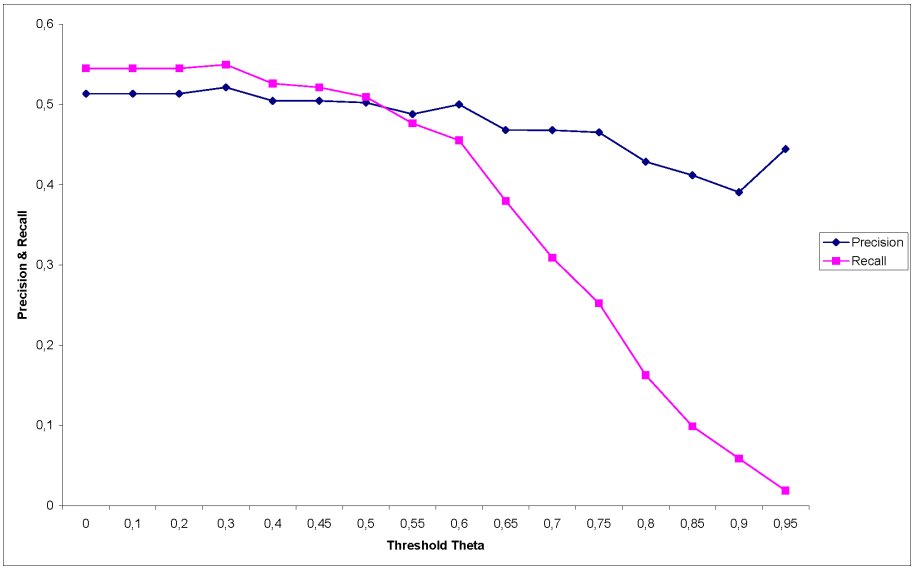

Figure 5.8: The curves show precision and recall values of the indices s and p when increasing the threshold θ. Precision means the proportion of pairs of languages added to the network that belong to the same language family. Recall gives the number of language pairs found in relation to the maximally possible pairwise relations when assuming a maximum of one link from a vertex to another ($\frac{|V|(|V|-1)}{2}$). Best F-scores of about 0.53 are found for $\theta = 0.4$.

5.5.3 Evaluating the Similarity Index s

We generated a series of LaPNets varying θ as described in the previous section. Further, we compared the resulting edges between languages to the genealogical classification the languages belong to. We computed precision (number of edges found as "genealogically correct" to the total number of edges in G_θ) and recall (number of edges found as "genealogically correct" to the total number of edges known as genealogically correct) for each LaPNet. Figure 5.8 shows the results. Precision and recall lie nearby 50%, that is, about 50% of the languages were linked correctly in genealogical terms only looking at phonological similarity. However, still 50% are linked incorrectly, and we would like to improve this result if this is indeed possible by means of phonological similarity.

5.5.4 Unifying the asymmetric similarity s

One disadvantage of s is its asymmetry. We would like to have a single value representing the similarity between two languages l_i and l_j. *Johannesson* (2000) presents a means to unify asymmetric similarity which we apply here to s. *Johannesson* (2000) introduces the *Relative Prominence Model* (RPM) which is defined as follows: given

[8]This condition can be omitted when we aim to examine not only the language most similar pair of languages but also other similarities among languages.

$s_{ij}, s_{ji} \neq 0$, he defines:

$$p_{ij} = \begin{cases} \mu(i,j)(s_{ij}/s_{ji}) & if\ s_{ij} < s_{ji}; \\ \mu(i,j)(s_{ji}/s_{ij}) & otherwise, \end{cases} \tag{5.4}$$

where μ is defined as $\mu(i,j) = (s_{ij} + s_{ji})/2$, and $p_{ij} = 0$ if $s_{ij}, s_{ji} = 0$.

Applying this model to LaPNet, we construct the networks as described in Section 5.5.2 modifying the algorithm as follows (the changes are highlighted in red):

Require: matrix T
Ensure: LaPNets $G_\theta(V,E)$ for $\theta = \{0, 0.05, ..., 1\}$
1: Set $\theta = 0$
2: Calculate the asymmetric similarity matrix S, the symmetric matrix P and the vector M of maximal values in each row in P
3: **while** $\theta \neq 1$ **do**
4: $E = \emptyset$
5: **for** $i = 1$ to $|L|$ **do**
6: **for** $j = 1$ to $|L|$ **do**
7: **if** $i \neq j \wedge\ p_{ij} > \theta \wedge p_{ij} == M_i$ **then**
8: add the edge e_{ij} to E
9: **end if**
10: **end for**
11: **end for**
12: print $G_\theta(V,E)$
13: $\theta = \theta + 0.05$
14: **end while**

Now, index p instead of s is compared with θ and the maximal value M_i. M is the vector of maximal values of P (instead of S). Consequently, we compare p_{ij} instead of s_{ij} and s_{ji} with the threshold, and if $p_{ij} > \theta$ we add an edge. According to *Johannesson* (2000), RPM should successfully match the corresponding asymmetric model. We construct the networks using the index p and compare the precision and recall values for p and s. In addition, we test an alternative version of RPM, taking $p_\mu = \mu(i,j)$. This is a simplification of the equation 5.4 which is computed as a possible alternative.

5.5.4.1 Evaluating RPM

Precision and recall values for the indices p and its variant p_μ are shown in Figures 5.9 and 5.10. The curves show that RPM improves the result from an *F-score* (i.e., harmonic mean between precision and recall) of 0.53 to 0.6. The variant with p_μ does not improve the F-score resulting in a decrease (F-score = 0.49). However, the overall behavior of μ is similar to RPM, although it does not achieve that high precision.

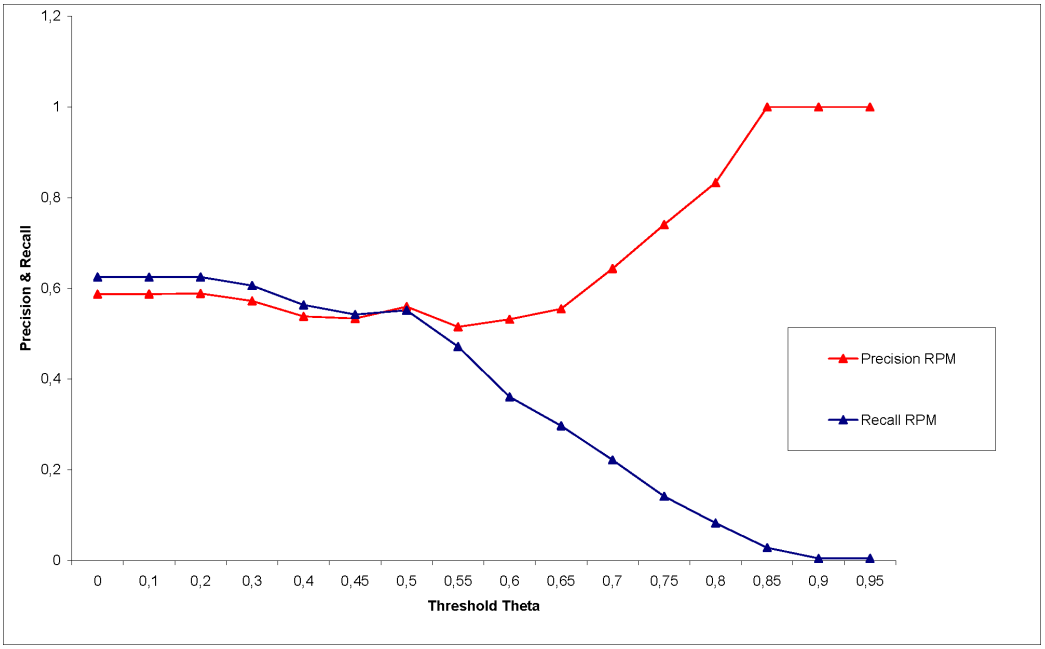

Figure 5.9: The curves show precision and recall values of the index p when increasing the threshold θ. Precision means the proportion of pairs of languages added to the network that belong to the same language family. Recall gives the number of language pairs found in relation to the maximally possible pairwise relations when assuming a maximum of one link from a vertex to another ($\frac{|V|(|V|-1)}{2}$). Best F-scores of about 0.6 are found for $\theta = 0$.

5.5.4.2 Discussion

Comparing the similarity indices s, p and p_μ, we can conclude that the RPM model performs best matching the most genealogical similarity between languages. For $\theta > 0.55$ in case of RPM, mostly genealogically related languages are linked. That means, languages with high phonological similarity (according to RPM) are related languages. Of course the higher θ the less language pairs have this high similarity which is comfirmed by the low recall in the upper ranks of θ. Optimal θ value, however, in terms of F-score seems to be at $\theta = 0$ for p and at $\theta = 0.4$ for p_μ. That means, these values of θ allow to include the maximal number of languages which are related. Interestingly both curves in Figure 5.10 have an inflection point at 0.55 and at 0.6. Up to this point precision and recall remain nearby constant. Starting from this inflection point, the curves diverge. Rising precision results in the fact that less languages are included which results in the overall decrease of recall. For index s both curves decrease with higher values of θ which means that only few related languages are found in general and even less when θ increases.

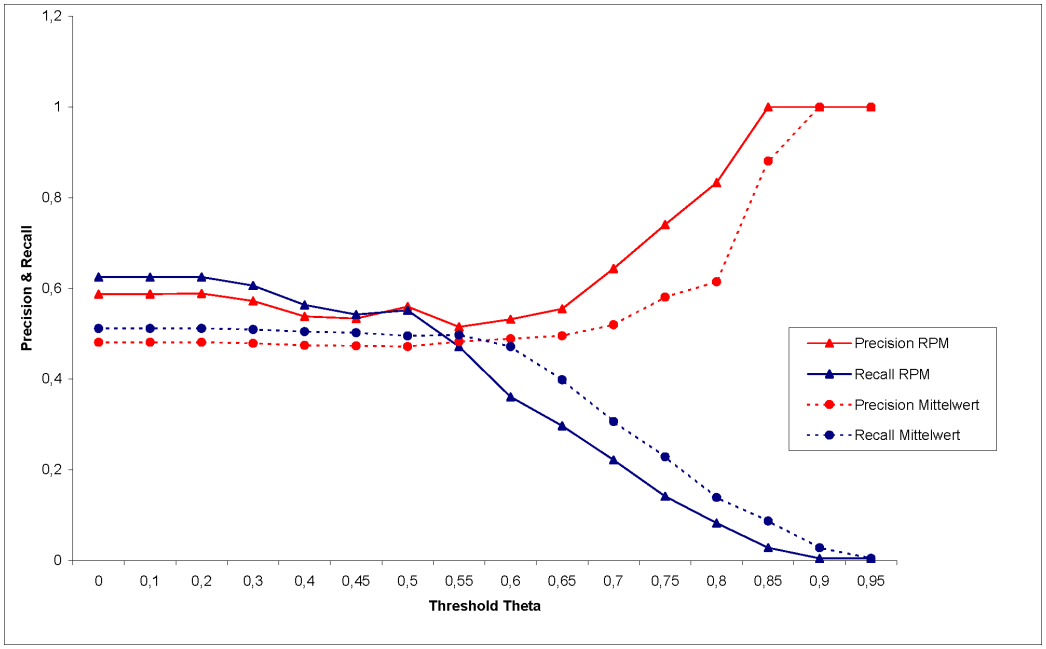

Figure 5.10: The curves show precision and recall values of the index p compared to its μ-alternative. Best F-score for p_μ of about 0.49 is found for $\theta \in [0; 0.3]$.

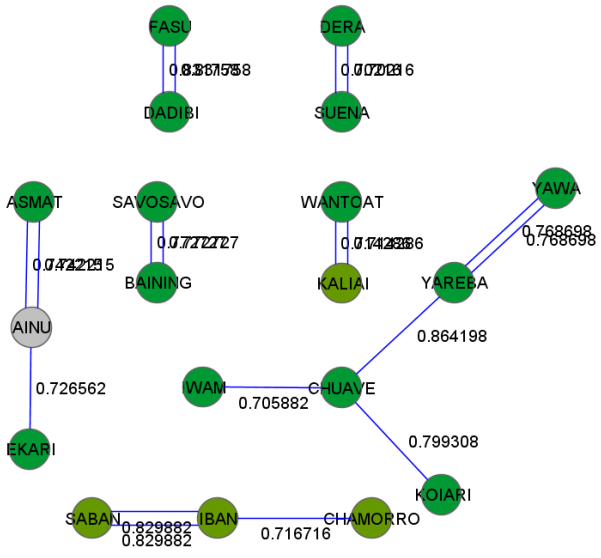

Figure 5.14: Austronesian (olive-green) and Papuan languages (green) and Ainu (grey).

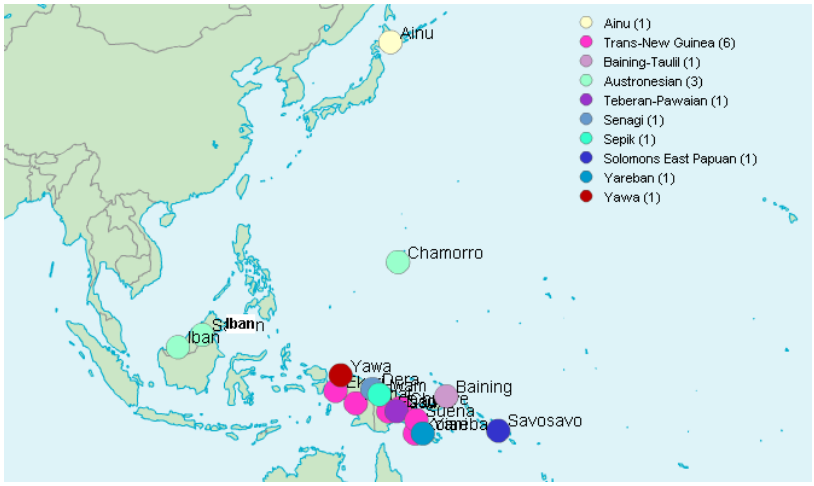

Figure 5.15: Austronesian and Papuan languages and Ainu – the geographical distribution (*Haspelmath et al.*, 2005).

5.5.5 Language Comparison by means of LaPNet

As shown in the previous section, the inflection point for RPM lies at $\theta \in [0.55; 0.6]$. Increasing θ improves the quality of links but reduces the number of languages in the network. Figure 5.11 shows the LaPNet produced by the p_μ-model for $\theta = 0.55$. Differently colored vertices represent languages belonging to different language families. Connected vertices of the same color represent languages belonging to the same family. The higher θ the smaller the network and the more vertices have the same color (i.e., the more languages that are phonologically similar are also related genetically. This is observable from LaPNets with $\theta \in [0.55; 0.7]$ in Figures 5.11-5.13. The networks are directed, which means that each language is connected to its most similar neighbor. The weights displayed in the figures represent the RPM-value among two languages.

The amount of languages connected that are phonologically similar and belong to the same family is about 60% (i.e., precision value) for $\theta < 0.55$, and $\sim 100\%$ for $\theta > 0.55$. When we look at the network produced considering $\theta = 0.7$, we get a much smaller network (Figure 5.13). However, most languages within this network are connected "correctly" in the sense of language family relationships. In addition they are mostly mutually connected, that is, language A is most similar to B, and B to A.

Now, we can zoom in and examine the individual components of the network. Look for instance at Figure 5.12; we can see that languages from Australian (yellow) and Niger-Kordofanian (dark blue) language families are most similar phonologically within the family. Furthermore, South American (turquoise) languages are also most strongly linked. Indo-European languages, in turn, are mostly dispersed although single languages are similar to their directly related sub-group neighbors (e.g., Russian and Bulgarian). Other pairs of the Indo-European group are not included in the networks of high similarity (i.e., with $\theta > 0.5$), although, they are similar to each

other but not to that large extent as, for example, the Australian languages.

Further, the Nilo-Saharan (pink) family is closely related to the Niger-Kordofanian. These are all African languages. From this example we see that phonological similarity can also exist between languages related geographically (see *Kapatsinski* (2008) who also observed this fact). The Nilo-Saharan language family is a group established by Joseph Greenberg who tried to unify genetically unclassified African languages. For the most of these languages there is an agreement among historical linguists that they belong to the single genealogical branch (i.e., a proto-language for these languages can be partially reconstructed). Some of them, however, are controversial, so that they could also belong to one of the other African language families (see *Bender* (1997); *Ehret* (2001)). This fact is also reflected in our networks - the members of the Nilo-Saharan family are pairwise related to the Niger-Kordofanian as well as to the Afro-Asiatic families. In sum, the problematic languages can be analyzed in isolation based on the method proposed here in order to verify their genetic relationship.

The last example we would like to discuss, is the language *Ainu*. Ainu is an isolated language spoken at the island *Hokkaidō* in Japan. Ainu is not related to Japanese, and comparative linguists classify Ainu to the group of Paleosiberian languages which unifies some languages that could not be classified. When we look at Figure 5.14 (an excerpt from the LaPNet $\theta = 0.7$, p-model), we see Ainu linked together with Papuan (green) languages, there are also Austronesian languages shown on this figure. Papuan and Austronesian languages are closely related geographically (see Fig. 5.15). In our LaPNet-representation, some of these languages (i.e., Papuan and Austronesian) are also linked together like, for example, Kaliai and Wantoat (see Figure 5.14). The question rises, why is Ainu connected to Papuan languages? *Tambovtsev* (2007) also studied Ainu, and he found out that according to his index the closest relative to Ainu is Tagalog, an Austronesian language. These findings lead to the assumption that Ainu could be related to the languages spoken in Indonesia. To prove this hypothesis, in addition to phonology, we need to compare these languages on other levels of linguistic representation.

5.6 Summary

In this chapter we have presented an approach to automatically classify languages by means of phonological information obtained from the UPSID. Moreover, we presented a network model that allows varying the similarity threshold to examine phonological similarities among languages and to relate them to their genealogical or typological relationships. We have found that some language families have preserved a high phonological similarity within the family (e.g., Australian), whereas other language families exhibit a high inner-family variability (i.e., Indo-European). However, the similarities among language sub-groups within individual families are mostly high, even for Indo-European. The findings indicate that changes languages undergo over time do neither completely eliminate their relationship to the language family nor to closely related languages. The factor of areal closeness can reinforce the phonological similarity as observed for some genetically related languages.

The presented network model can be used to verify the formation of a language family. For some languages (there are 120 isolated languages in the world) the method can be applied in order to classify them to existing families or to reclassify them to another family. Controversial languages in a language family (e.g., parts of the Nilo-Saharan family) can also be tested this way.

In summary, concerning the questions posed at the beginning of the chapter, we can say that phoneme inventories can lead to essential conclusions about the language family a language belongs to. We could distinguish between language families looking solely at the phoneme inventories of the corresponding languages. Genealogical distances between single languages can be partially reconstructed based on phonology; however, further research is needed to evaluate the expressiveness of phonological distance between languages of different families.

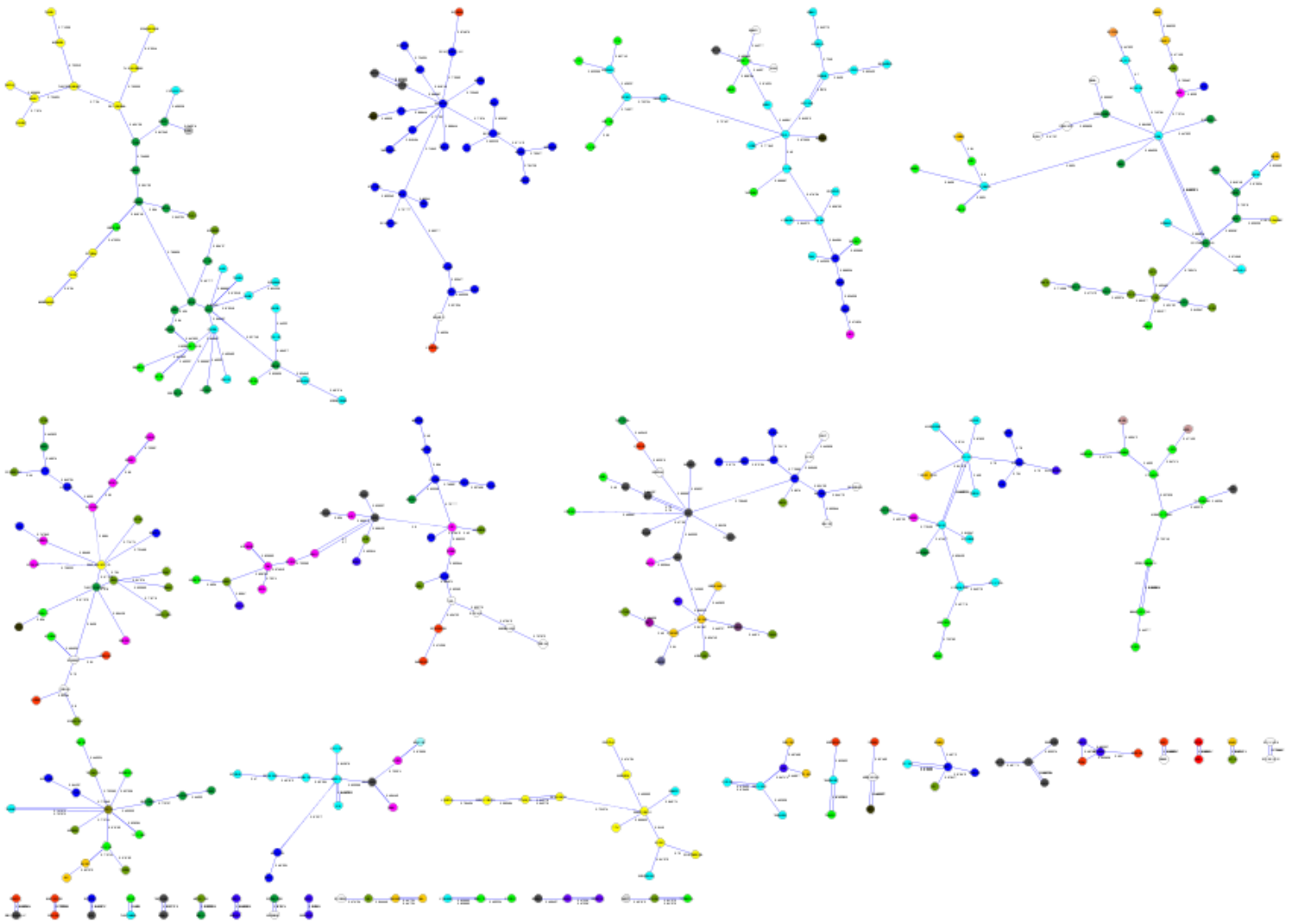

Figure 5.11: LaPNet induced by the p_μ-model for $\theta = 0.55$. Different colors represent different language families. As can be seen from the figure, all connected vertices sharing the same color belong to the same language family. In these cases, language family relationships are recognized by means of phonological similarity.

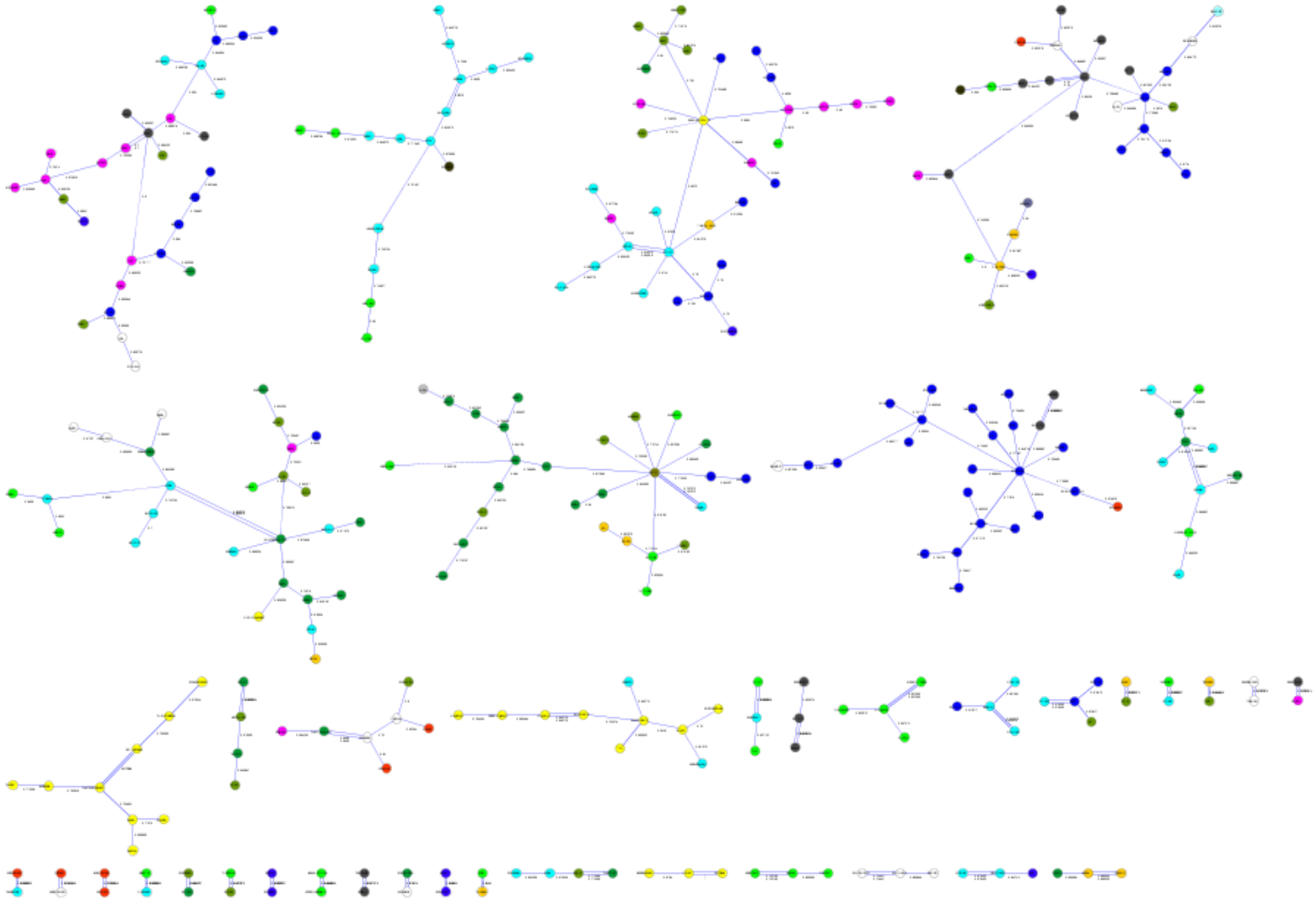

Figure 5.12: LaPNet induced by the p-model for $\theta = 0.6$. Different colors represent different language families. As can be seen from the figure, all connected vertices sharing the same color belong to the same language family. In these cases, language family relationships are recognized by means of phonological similarity.

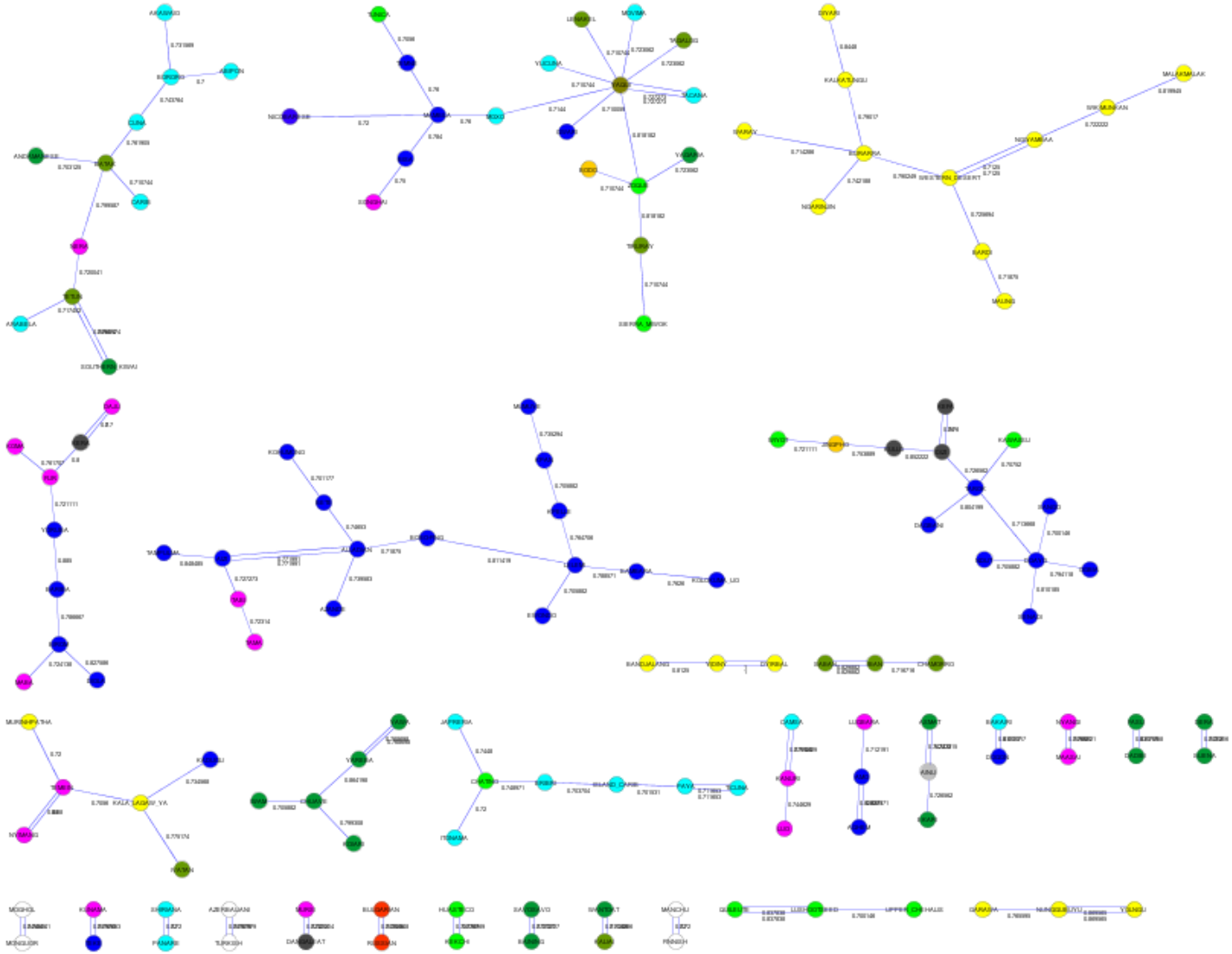

Figure 5.13: LaPNet induced by the p-model for $\theta = 0.7$. Different colors represent different language families. As can be seen from the figure, all connected vertices sharing the same color belong to the same language family. In these cases, language family relationships are recognized by means of phonological similarity.

CHAPTER VI

Syntactic Dependency Networks

6.1 Introduction

In this chapter, we will deal with syntactic networks. In general, there are several possibilities to define a syntactic network. We adapt the notion of a Global Syntactic Dependency Network (GSDN) from *Ferrer i Cancho et al.* (2004), which is constructed from syntactic *dependency treebanks*. Treebanks represent an indispensable resource in the area of computational linguistics and natural language processing (NLP) that enable researchers to train and test syntactic parsers and evaluate NLP applications (*Nivre*, 2005). Section 6.2 gives a short overview on syntactic theories and promotes the use of dependency treebanks for cross-language comparison as intended in the present work. Section 6.3 discusses the problem of heterogeneity of available treebanks on different levels of comparison. In Section 6.5, we give an overview of the treebanks used in this thesis focussing on their specifics. In Section 6.6, we present the definition of GSDNs adapted from *Ferrer i Cancho et al.* (2004) and describe how we induce the networks from dependency treebanks. Finally, the content of the chapter is summarized in Section 6.7.

6.2 Selecting the appropriate Syntactic Framework

Our goal is to find a means of constructing a syntactic network for different languages from natural language data. Firstly, we have to find a theoretic framework that divides language into appropriate units linking them together by means of syntactic relations. This framework should serve as a base for construction of the network. Secondly, we need corpora of natural language data (written or spoken) annotated with syntactic information (i.e., *treebanks*). In this section we review the two main approaches to syntax widely used for syntactic representation of language - *constituency* and *dependency*.

6.2.1 Constituency

The constituency based approach to syntax gained attention in the 50ies of the last century. *Bloomfield* (1933) proposed the constituency analysis inspired by the struc-

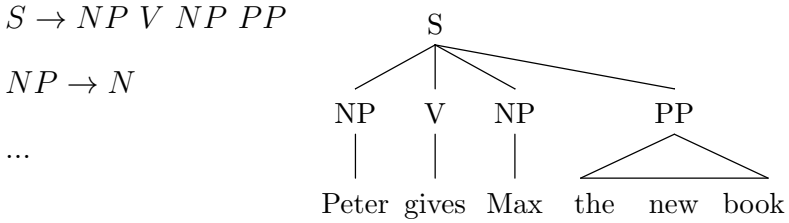

Figure 6.1: A Sentence analyzed with constituency structure.

turalist tradition. *Chomsky* (1957) formalized the approach "in the model of phrase structure grammar, or context-free grammar" (*Nivre*, 2006, 10). In this model (we will use the term Phrase-based Grammar (PG) henceforth[1]) texts (or spoken data) are analyzed recursively: words are grouped together in phrases, phrases in clauses, clauses in sentences resulting in a syntactic tree. Phrases are denoted by the governing element, for example *noun phrases*, *verb phrases* etc.[2] Using this apparatus, a language can be described by means of a finite set of rewrite rules that allow to produce an infinite number of utterances. However, natural languages posit a challenge to the formalism resulting in ambiguities, multiple interpretations, discontinuity, etc. An example sentence in PG is shown in Figure 6.1. The sentence "Peter gives Max the new book" contains non-terminal and terminal nodes and is rewritten by means of recursive rules such as $S \rightarrow NP...$, $NP \rightarrow N$, etc.

Based on this model, various syntactic theories (e.g., Generalized Phrase- Structure Grammar (GPSG) (*Klein and und Geoffrey Pullum*, 1985), Lexical Functional Grammar (LFG) (*Kaplan and Bresnan*, 1982), Head-driven Phrase-Structure Grammar (HPSG) (*Pollard and Sag*, 1994) etc.[3]) emerged in support of (automatic) natural language processing. All these theories ground on the assumption that syntactic relations are "part-of-the-whole" relations between words and phrases, putting phrases on the same scale with words (*Hudson*, 1994, 90). As a result of this development, different syntactically annotated treebanks for various languages were made available lately.

The advantage of this approach is that a small finite number of rules suffices to produce an infinite number of sentences of a language. However, since the linear order of the sentence plays a crucial role in determining the sentence structure the PG is more suitable for languages with fixed word order (like English) than for free word order languages.

6.2.2 Dependency

Although there is some evidence that the dependency tradition can be traced back to antiquity (Kruijff, 2002)[4], dependency grammar (DG) started to gain impor-

[1] We refer to *Hudson* (1994), who distinguished between the *phrase-based grammar* (PG) and *dependency-based grammar* (DG).

[2] *Nivre* (2006).

[3] See *Nivre* (2006) for more details.

[4] *Nivre* (2006, 11).

tance through the famous work of Lucien Tesnière "Eléments de syntaxe structurale" (*Tesnière*, 1959). DG starts from the assumption that words in a sentence are connected (apart from (local) phrasal relations) by means of semantic relations or '*ordre structural*'. The lexical semantics of a word determines its position in the tree. The predicate of a sentence (e.g., verb) that occupies the root position defines the selection of appropriate arguments. Other elements (direct / indirect objects) directly depending on the predicate become its immediate daughter nodes. Articles, modifiers, etc. are subordinated to the arguments they specify. Non-terminal nodes (e.g., clauses, phrases) are not part of the syntactic hierarchy. Only word-to-word relations irrespective of the position of a word are considered. That means, the linear order of words in the sentence can be (!) discarded in this approach. This flexibility makes the DG less restrictive and better applicable to a wide range of languages (e.g., free word order languages) since word-to-word relations hold independently from the position of a word.

However, the connections between the elements in the tree are subject to several conditions restricting the flexibility of the approach to some extent. These restrictions were made in order to demarcate grammatical sentences from ungrammatical ones. The restrictions in form of axioms are formulated by *Robinson* (1970) as follows:

1. one and only one element is independent (*single-head constraint*);
2. all others depend directly on some element (*connectedness constraint*);
3. no element depends directly on more than one other (*uniqueness constraint*);
4. if A depends directly on B and some element C intervenes between them (in linear order of string), then C depends directly on A or on B or on some other intervening element (*projectivity constraint*).

The first two axioms are widely regarded as uncontroversial constraining the trees to have a *single-head* and to be *connected*. The third axiom does not permit multiple heads.[5] The last axiom stipulating *projectivity* gives much more reason for debate - it does not allow crossing edges with respect to the linear order of words in a sentence and this constraint seems to be hardly tractable in practice.Dependency theories used in NLP mostly abandon the fourth constraint while retaining the first three (except for, e.g., *Hudson* (1984), who omits the *uniqueness* constraint allowing multiple heads and cycles).

To overcome the problem of non-projectivity, some dependency theories presume multiple levels - one abstract level with non-crossing branches, and several more concrete levels allowing discontinuity (e.g., Functional Generative Description (FGD) (*Sgall et al.*, 1986), Meaning-Text Theory (MTT) (*Mel'čuk*, 1988) to mention some). Others enrich the dependency structure with additional links relaxing the *uniqueness* constraint (Word Grammar (WG) (*Hudson*, 1984)). Yet, others separate the sentence's surface structure from its dependency representation .[6]

[5]See *Hudson* (1994) for a discussion why this constraint is problematic for empirical applications.
[6]See *Debusmann* (2000) for a comparison.

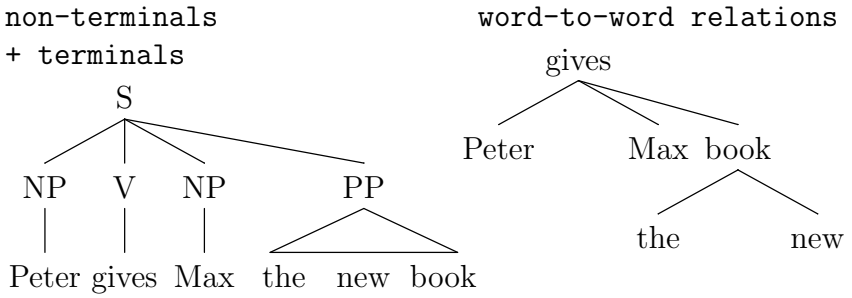

Figure 6.2: A Sentence represented using PG (left) and DG (right).

6.2.3 Summary

In summary, PG and DG offer two distinct perspectives on the structure of a language. The PG concentrates on the surface-structural relations dividing a sentence into phrases, whereas DG is concerned with lexical dependency relations among words. Both approaches are widely used for parsing of texts and there is a large number of treebanks annotated by means of either of the grammars. Some attempts to combine both approaches in order to achieve a more adequate representation of a language have been made, for example, in terms of the HPSG (*Pollard and Sag*, 1994).[7]

For the purpose of the present study it was important to find a framework applicable to as many diverse languages as possible allowing for comparative investigations. We used 17 different treebanks annotated with the DG (see Sec. 6.5 for the description of treebanks). The treebanks were annotated independently of each other by different research projects that all used the DG for data representation. Of course, DG served as the general framework for all of the 17 treebanks, however, each language has its own specifics and hence, individual adaptations of the DG were performed in order to make it suitable for each particular language. This additional diversification can bias quantitative comparisons of these treebanks. The following section discusses some issues that are to be taken into account when comparing treebanks developed under different theoretical assumptions.

6.3 Treebanks – Levels of Diversification

In order to induce syntactic networks, we needed a number of dependency treebanks – at least one for every language. Fortunately, there is a large number of dependency treebanks for a wide range of languages.[8] However, these treebanks differ in many respects that posits a challenge to the comparison of languages based on these data. *Pustylnikov and Mehler* (2008) outline three levels of diversification that can occur when comparing different treebanks.

[7]See also *Teich* (1998) for more issues on PG, DG and combined methods in parsing and syntactic representation.

[8]See *Kakkonen* (2005) for a review on existing treebanks.

- Level 1: refers to the *corpus genre*
- Level 2: relates to the *annotation theory*
- Level 3: relates to the *representation format*

On the first level, treebanks can be distinguished by means of the linguistic theory underlying the treebank creation, for example, PG vs. DG. The second level refers to the annotation theory that guided the annotators of the treebanks, for example, WG vs. FGD (both have the same Level 1 - the dependency theory). On the third level treebanks can vary according to the annotation format used to represent the data, that is, the same treebank can be represented in CoNLL-X (*Sang and Buchholz*, 2000), PENN (*Marcus et al.*, 1993) or other formats.

While pre-processing the treebanks, we aimed to eliminate the maximum of differences between the treebanks on all three levels. The diversity on the first level of corpus genre was resolved by selecting treebanks structured with respect to a particular syntactic theory (see. Sec. 6.2). As mentioned in the previous section we concentrate on the syntactic framework of dependency grammar here only.

Diversity on the third level (annotation format) was also resolved by transforming all 13 treebanks into a unified representation format. Attempts to provide an exchange format for treebanks were made by *Sang and Buchholz* (2000) as part of the *CoNNL-X shared task*[9] or by TIGER-XML (*Mengel and Lezius*, 2000). CoNNL-X is a simple text based format where each word of a sentence (+ its tab-separated attributes) constitutes a line and sentences are separated by a blank line. This format is widely used due to its simplicity and minimal parsing costs. Other formats such as, for example, TIGER-XML (*Mengel and Lezius*, 2000), eGXL (*Pustylnikov and Mehler*, 2008) are often preferred since they support the interoperability of data by means of XML. However, they are more complex and require a higher adaptation effort. For the purpose of our work, we decided to make use of the DTDB (*Pustylnikov et al.*, 2008) which integrates 17 treebanks by means of the eGXL format.[10]

The second level of diversification was the most difficult to bridge. Level 2 concerns differences emerging from the use of a different annotation theory when creating a treebank. In our case, we deal with different dependency theories within the *dependency grammar family*[11] that guided the annotation process of our treebanks. These differences could have had a substantial influence on the resulting networks and so it was in our interest to identify and, if possible, to eliminate them. As we will see later, some differences can not be eliminated without modifying the syntactic structure. In these cases, it is important to keep them in mind when interpreting the results of the language classification. In the following sub-sections, we outline the major differences attributed to particular dependency theories that become important when we will introduce the single treebanks. In the subsequent section we examine whether the critical aspects outlined here hold for our treebanks.

[9] `http://nextens.uvt.nl/~conll/`.

[10] See `www.treebankwiki.org` for an overview on treebanks included into DTDB.

[11] The term is adapted from *Hudson* (1994).

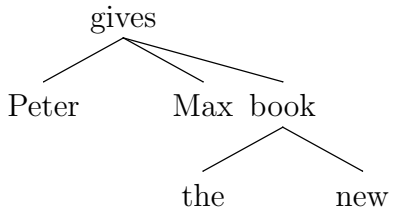

Figure 6.3: The Sentence: *"Peter gives Max the new book."* in DG notation.

6.3.1 Coordination

The first problem we want to address is the representation of coordinated structures in DG. The dependency representation of a sentence is based on relations from a head to a dependent, whereby the role of a particular head (or dependent) depends on the lexical semantics of the word. The dependency hierarchy starts, as usual, with a predicate (e.g., a verb) that selects its dependent arguments. That is, the valency of the verb determines the number of arguments. For example, the verb *give* requires 3 arguments as in the sentence (Fig. 6.3) "Peter gives Max a book", *Peter*, *Max* and *book* are the arguments, and also the immediate daughter nodes of *gives*. On lower levels of dependency, elements that modify the arguments are attached to them. In the above example, *the* and *new* modify *book*, thus, the dependency (relation) branching out from *book* is assigned to them.[12] That way, the dependency tree can be constructed. Problems arise when we have coordinated constructions such as, "Peter and Jane give Max the new book". There is no agreement amongst theories how to represent the coordinated constructions. Clearly, *Peter and Jane* represent the subject of the sentence and are dependent on *give*. However, it is difficult to describe the relation between *Peter* and *Jane* in terms of dependencies (*Nivre*, 2006). Some theories (e.g., FGD) solve the problem by treating the coordinating conjunction *and* as a head of *Peter* and *Jane*, others treat the first conjunct as the head dominating the conjunction, which, in turn, dominates the second conjunct (as in MTT (*Mel'čuk*, 1988)). Yet another option is to substitute the dependency representation for some sort of phrase structure (WG (*Hudson*, 1984)) representing *Peter and Jane* as a single phrase and, in turn, the single subject to *give*. Finally, the analysis of *Tesnière* (1959) treats both conjuncts as depending on the head-word *give*.[13] Coordination also violates the constraint of projectivity producing discontinuous constructions (*Hudson*, 1994).

The problem with coordination is solved differently in the various dependency theories, thus, when we aim to compare treebanks annotated with different formalisms the same constructions may be represented differently so we have to take these differences into account when interpreting the results.

[12]Different theories place the direction of the arc either from the head to the dependent or vice versa. There is no general agreement on that (*Nivre*, 2006). We pursue the convention to place the arc from the head to the dependent. We convert treebanks to this convention, if they encode the arc the other way around.

[13]See *Nivre* (2006, 54-55).

6.3.2 Punctuation

Similar to coordination, treebanks vary in the way they deal with punctuation marks and other special symbols. Some of the treebanks (e.g., the Prague Dependency Treebank (PDT) or others - see Tab. 6.1) use punctuation marks as nodes of the dependency hierarchy. Other treebanks do not. This small difference can add at least one additional node and one link to the sentence. Sometimes it changes the dependency structure when the punctuation mark itself is permitted to govern other elements. Statistically, the inclusion of punctuation can strongly influence the structure of the sentences on the large scale as well as the resulting network.

A unification of treebanks is not straightforwardly possible without loss of information since punctuation marks can govern other elements that have to be *rewired* first before removing the mark. In order to eliminate this bias, we removed all punctuation marks and other symbols from the dependency representation. Governed elements were rewired to depend on the head element of the punctuation mark.[14]

6.3.3 Projectivity

The last difference we like to mention here concerns the *constraint of projectivity*. This constraint concerns the linear order of words in a sentence. A dependency sentence can be represented as a *projective graph* if it fulfills the following conditions according to *Nivre* (2006, 71):

Single-Head-Constraint Every node has maximally one head, if $i \rightarrow j$ then there is no node k so that $k \neq i$ and $k \rightarrow j$.

Acyclicity-Constraint The graph G is acyclic, if $i \rightarrow j$ then not $j \rightarrow^* i$.

Projectivity-Constraint The graph G is projective, if $i \rightarrow j$ then $i \rightarrow^* k$, for every node k so that $i < k < j$ or $j < k < i$.

Thus, according to the last constraint a word k appearing in between the head $i(j)$ and the dependent $j(i)$ must depend on the head. Theories like MTT (*Mel'čuk*, 1988) or WG (*Hudson*, 1984) allow so called *long-distance dependencies* or *discontinuous trees* to violate the constraint since they are assumed to occur in free word order languages. However, the analysis of *Havelka* (2007) on eleven language treebanks indicates that projectivity is very rare across languages occurring seldom, even in free word order languages such as Czech. On the contrary, most of the sentences are well nested.

In the present work, projectivity does not impose any difficulties since the networks we construct do not require projective structures.

[14]This is of course an abstraction and a deviation from the original formalism. However, we decided to perform this step in order achieve a better comparability of the data.

6.4 Dependency Theories Used

6.4.1 Treebanks developed by means of Functional Generative Description (FGD)

This section deals with treebanks annotated using the Functional Generative Description (*Sgall et al.*, 1986) that represent the majority of our treebank database. The FDG is a stratificational or multi-level approach to representing a sentence that allows for up to four levels of functional annotation: morphematical, morphonological, analytical (surface syntax) and tectogrammatical (deep syntax) levels. The FDG was developed by Petr Sgall and his research group in the 1960s in Prague following the tradition of *Tesnière* and the Prague linguists. Treebanks annotated with FDG based formalisms do not impose the constraint of projectivity which is an important assumption when dealing with, for example, Slavic languages. Furthermore, punctuation marks can be included into the dependency hierarchy, however, not all the formalisms do (e.g., Russian). To represent coordinated constructions, most formalisms focussed on here treat the coordinating conjunction (or a punctuation mark) as a head of the coordinated words.

6.4.2 Treebanks relying on HPSG

The Head-driven phrase structure grammar (HPSG) was developed by *Pollard and Sag* in the 1980s as a unification grammar that allows to store any sort of linguistic information (syntactic, phonological, etc.) in a single hierarchically organized attribute-value matrix. Here, only a single level of representation is used that contains all the information on the word including its morpho-syntactic features, head-dependence information, valency, etc. In contrast to CG there are no rules for binding or movement of constituents - the grammar is built in terms of restrictions that are expressed via the corresponding lexical items. If a noun and an adjective exhibit agreement, the corresponding feature is added to both items.

Treebanks originally having been developed using the HPSG theory are Bulgarian, Dutch and Japanese. They were converted from HPSG into dependency based representations. Their statistical properties are discussed in the subsequent sections.

6.4.3 Treebanks based on Word Grammar (WG)

Word grammar is a theoretical framework to describe language structure that was developed by *Hudson* (1984) in the 1980s. WG views language as an inheritance-network or as a dependence hierarchy. The representation is monostratal, that is, morphological, syntactic, semantic and conceptual information can be encoded in terms of hierarchical relations on the lexical items, as nodes. Thus, multiple inheritance as well as discontinous and non-projective constructions can occur. Coordination represents an exception in WG since coordinated constructions are treated as a complete phrase, whereby external relations apply to the whole coordinated conjunctions rather than to the single conjuncts. However, different annotation frameworks

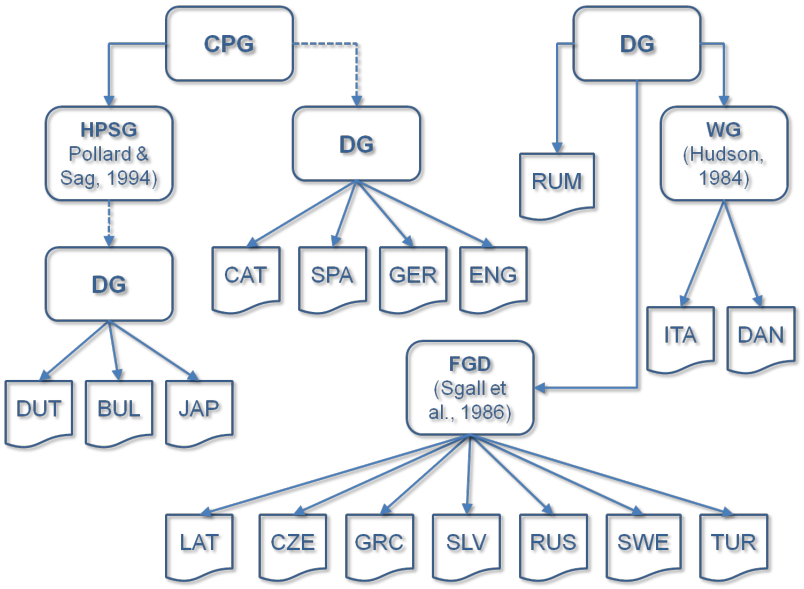

Figure 6.4: Classification of treebanks according to the dependency theory used for annotation. CPG - constituent phrase-structure grammar, DG - dependency grammar, FGD - Functional Generative Description, HPSG - Head-driven Phrase Structure Grammar, WG - Word Grammar. Theory independent treebanks are directly attached to the DG node (Romanian). Treebanks that are theory independent but were converted from CPG have the CPG node as a root (CPG > DG > CAT, SPA, etc.). Dashed lines represent conversion processes.

deal differently with coordination. In Danish and Italian treebanks (i.e., the two WG treebanks of our sample) the first conjunct dominates the conjunction and the conjunction dominates the second conjunct (e.g., "apples" → "and" → "pears").

6.5 Data: Dependency Treebanks Used

In this section we look more closely at the properties of analyzed treebanks with respect to the diversification criteria discussed in the previous sections. The general properties of the treebanks are listed in Table 6.1.[15] As can be seen from Figure 6.4, seven treebanks use FGD as the underlying dependency framework. However, as becomes evident from subsequent sections, even those treebank differ much in the realization of the formalism adapted to match the peculiarities of individual languages (e.g., with respect to *coordination*). Also seven treebanks are derived from the CPG; three of them rely on the HPSG theory and four are theory independent. Regarding *punctuation*, there are three treebanks that do not include punctuation marks within the dependency tree - Russian (FGD), Romanian (DG) and Italian (WG).

[15]The language abbreviations used correspond to the ISO 639 Language Codes norm (`www.w3.org/WAI/ER/IG/ert/iso639.htm`).

treebank	abbrv.	language	punct. heads	punct. included	reference	format used
BulTreeBank	BUL	Bulgarian	no	yes	*Osenova and Simov* (2004)	CoNNL
CESS - Catalan Dependency Treebank	CAT	Catalan	no	yes	*Civit et al.* (2004)	CoNLL
Prague Dependency Treebank 2.0	CZE	Czech	no	yes	*Hajič* (1998)	PDT
Danish Dependency Treebank v. 1.0	DAN	Danish	yes	yes	*Kromann* (2003)	TIGER-XML
Alpino Treebank v. 1.2	DUT	Dutch	no	yes	*van der Beek et al.* (2002)	CoNLL
2008 CoNLL Shared Task Data	ENG	English	no	yes	*Surdeanu et al.* (2009)	CoNLL
TIGER-DB	GER	German	no	yes	*Brants et al.* (2002)	CoNLL
Ancient Greek Treebank	GRC	Greek	yes	yes	*Bamman and Crane* (2006)	XML
Turin University Treebank v. 0.1	ITA	Italian	yes	no	*Bosco et al.* (2000)	TUT format
VERBMOBIL Japanese Treebank	JAP	Japanese	no	yes	*Hinrichs et al.* (2000)	CoNLL
Ancient Latin Treebank	LAT	Latin	yes	yes	*Bamman and Crane* (2006)	XML
Sample of sentences of the					`http://www.phobos.ro/`	
Dependency Grammar Annotator	RUM	Romanian	no	no	`roric/DGA/dga.html`	simple XML
Russian National Corpus	RUS	Russian	no	no	*Boguslavsky et al.* (2002)	RNC-XML
A sample of the Slovene						
Dependency Treebank v. 0.4	SLO	Slovene	yes	yes	*Džeroski et al.* (2006)	TEI
Cast3LB - Spanish Dependency Treebank	SPA	Spanish	yes	yes	*Civit and Martí* (2005)	CoNLL
Talkbanken05 v. 1.1	SWE	Swedish	yes	yes	*Nivre et al.* (2006)	TIGER-XML
METU Sabanci Treebank	TUR	Turkish	yes	yes	*Oflazer et al.* (2003)	CoNLL

Table 6.1: Treebanks sorted in alphabetical order of language names. The column **punct. heads** indicates whether punctuation marks can function as heads of other elements or not. The column **punct. included** indicates whether punctuation marks are included into the dependency tree or not.

6.5.1 Alpino Dependency Treebank

6.5.1.1 General Characteristics

	Description
size:	$\sim$ 172.000 tokens, 13.349 sentences
text types:	newspaper part of the Dutch *Eindhoven corpus* (*den Boogaard*, 1975)
dependency structure:	theory independent
annotation formalism:	based ont the spoken CGN corpus (*Oostdijk*, 2000)
	and on Tiger Treebank (*Skut et al.*, 1997).
annotation formats	xml, conll

Table 6.2: Characteristics of the Dutch treebank.

6.5.1.2 Description

The Alpino treebank is annotated with a theory independent dependency structure. No multiple heads are used, the second head is subordinated to the main head word (e.g. a finite verb - to the auxiliary verb). Coordinated conjuncts are either subordinated to the head or to the coordinating conjunction but not to the punctuation mark. Punctuation marks are included into the dependency structure but only

as subordinates and not as heads. Thus, punctuation marks can be removed from dependency trees without losing the entire dependencies between the words.

6.5.2 Bulgarian BulTreeBank

6.5.2.1 General Characteristics

	Description
size:	$\sim$ 196.000 tokens, 12.823 sentences
text types:	1.500 sentences from Bulgarian grammars and 10.000 newspaper, government document and prose texts (*Osenova and Simov*, 2004)
dependency structure:	HPSG language model transformed to dependency annotation
annotation formalism:	HPSG (*Pollard and Sag*, 1988)
annotation formats	conll

Table 6.3: Characteristics of the Bulgarian treebank.

6.5.2.2 Description

The BulTreeBank-DP, that is, the BulTreeBank transformed into dependency annotation comprises 1.500 sentences from Bulgarian grammar textbooks as well as 10.000 sentences from newspapers, literature and legal documents. Therefore we are dealing with a corpus of heterogenous genres, here. The annotation scheme for Bulgarian was developed based on the HPSG-framework. The corpora were preprocessed automatically (morpho-syntactically, POS, NPs) and annotated manually with syntactic structure. Punctuation is included into the dependency tree but only as a "leaf" (no head-punctuation marks). That makes it easy to delete punctuation marks when unifying the trees in order to build a GSDN. Coordinated conjunction and the second conjunct are subordinated to the first conjunct. Long distance dependencies and discontinuous trees can occur in the treebank.

6.5.3 Catalan Cat3LB Treebank

6.5.3.1 General Characteristics

	Description
size:	$\sim$ 478.000 tokens, 16.631 sentences
text types:	news wire
dependency structure:	theory neutral CPG, transformed to dependency annotation
annotation formalism:	surface-oriented annotation
annotation formats	conll

Table 6.4: Characteristics of the Catalan treebank.

6.5.3.2 Description

The Catalan dependency treebank CAT3LB was developed by (*Civit et al.*, 2004). The treebank represents a semi-automatic conversion of a constituency based treebank to dependency (automatic conversion with a hand made table of head relations). The treebank contains discontinuous constructions which are marked in the annotation. Coordinated conjunction and the second conjunct are subordinated to the first conjunct. Punctuation marks are attached to the head of the sentence and do not function as heads.

6.5.4 Spanish Cast3LB Treebank

6.5.4.1 General Characteristics

	Description
size:	125.000 tokens, 3.512 sentences
text types:	newspapers, novels, scientific papers, etc.
dependency structure:	theory neutral CPG, transformed to dependency annotation
annotation formalism:	surface-oriented annotation
annotation formats	conll

Table 6.5: Characteristics of the Spanish treebank.

6.5.4.2 Description

The Spanish dependency treebank Cast3LB (*Civit et al.*, 2004) was semi-automatically converted from constituency to dependency using the same approach as in the case of Catalan. The specifics of syntactic representation are the same as for Catalan, except for the punctuation marks which can govern other elements.

6.5.5 Romanian Dependency Treebank

6.5.5.1 General Characteristics

	Description
size:	$\sim$ 36.150 tokens, 4.042 sentences
text types:	newspaper articles
dependency structure:	dependency grammar
annotation formalism:	dependency formalism for Romanian
annotation formats	simple XML

Table 6.6: Characteristics of the Romanian treebank.

6.5.5.2 Description

The Romanian dependency treebank is constructed from newspaper articles. Punctuation is not included into the annotation. Discontinuous constructions are not annotated. Coordinated conjunctions dominate both conjuncts in a coordinated construction.

6.5.6 Italian Turin University Treebank

6.5.6.1 General Characteristics

	Description
size:	∼ 41.544 tokens, 1.500 sentences
text types:	newspaper, civil law
dependency structure:	dependency grammar
annotation formalism:	based on Word Grammar (*Hudson*, 1984)
annotation formats	TUT-format, TUT-Penn, conll

Table 6.7: Characteristics of the Italian treebank.

6.5.6.2 Description

The Italian dependency treebank is compiled from newspaper texts and civil law. Syntactic annotations are made semi-automatically by means of a parser and human supervision. Discontinuous constructions are annotated by means of trace elements that mark raising of elements, pro-drops, etc.. Coordinated conjunctions are governed by the first conjunct and dominate the second conjunct in a coordinated construction. Punctuation marks are part of the syntactic tree, they can also function as a head.

6.5.7 Czech Prague Dependency Treebank

6.5.7.1 General Characteristics

	Description
size:	∼ 1.290.000 tokens, 88.374 sentences
text types:	scientific, daily, business newspapers and journals
dependency structure:	dependency grammar
annotation formalism:	Functional Generative Description (FGD) (*Sgall et al.*, 1986)
annotation formats	PML-XML, FS, CSTS

Table 6.8: Characteristics of the Czech treebank.

6.5.7.2 Description

The Prague dependency treebank is compiled from various types of newspapers and journals. The syntactic annotation was performed manually by a team of six annotators. In later stages, support was provided by automatically generating trees and presenting them to the judgement of the annotator. Discontinuous constructions and punctuation are included into the tree. Punctuation marks function only as dependents and never as heads. Coordinated conjunctions govern both conjuncts in a coordinated sentence.

6.5.8 Russian Dependency Treebank

6.5.8.1 General Characteristics

	Description
size:	$\sim$ 256.800 tokens, 17.628 sentences
text types:	fiction, newspaper, scientific, short stories from internet (political,
financial,	cultural, sports news, hi-tech)
dependency structure:	dependency grammar
annotation formalism:	based on Functional Generative Description (FGD) (*Sgall et al.*, 1986)
annotation formats	TEI-XML

Table 6.9: Characteristics of the Russian treebank.

6.5.8.2 Description

The Russian dependency treebank represents the syntactically annotated part of the Russian National Corpus kindly provided by Leonid Iomdin and his research group (*Boguslavsky et al.*, 2002). The syntactic annotation comprises 78 syntactic categories developed for Russian (in contrast to Prague Dependency Treebank with 23 relations). The treebank is annotated semi-automatically by means of preprocessing by machine and human post-editing. Punctuation (though preserved as text in the TEI-annotation) is not included into syntactic tree. Missing verbs in copulative constructions or ellipses are reconstructed inserting so called 'phantom' elements.[16] Discontinuous constructions can occur. Coordinated conjunctions mostly depend on the first conjunct and govern the second.

	Description
size:	∼ 30.000 tokens, 1.998 sentences
text types:	prose
dependency structure:	dependency grammar
annotation formalism:	surface-syntactic, adapted from PDT (*Sgall et al.*, 1986)
annotation formats	TEI P4-XML, conll

Table 6.10: Characteristics of the Slovene treebank.

6.5.9 Slovene Dependency Treebank

6.5.9.1 General Characteristics

6.5.9.2 Description

The Slovene dependency treebank is compiled from the part of the morpho-syntactically annotated Slovene part of the parallel MULTEXT-East corpus (i.e. the first third of the Slovene translation of the novel "1984" by G. Orwell).[17] The treebank is annotated semi-automatically by means of mechanical pre-processing and human post-editing. Punctuation can function as head and as dependent. Discontinuous constructions can occur. Coordinated conjunctions depend on the first conjunct and govern the second.

6.5.10 Danish Dependency Treebank

6.5.10.1 General Characteristics

	Description
size:	∼ 100.000 tokens, 5.512 sentences
text types:	different types of newspaper, journals, etc.
dependency structure:	dependency grammar
annotation formalism:	Discontinuous Grammar (*Kromann*, 2003), based on Word Grammar (*Hudson*, 1984)
annotation formats	TIGER-XML, conll

Table 6.11: Characteristics of the Danish treebank.

6.5.10.2 Description

The Danish dependency treebank contains a random sample of texts taken from the PAROLE-DK[18], a balanced corpus of written Danish. The treebank is annotated and corrected manually. Punctuation can function as head and as dependent. Discontinuous constructions occur, there are three different kinds of dependency relations

[16] In our study we do not consider phantom elements constructing the network. The reason is that we take the language as it was produced in order to avoid biases due to syntactic enrichments made.

[17] See `http://nl.ijs.si/sdt/`.

[18] `http://ordnet.dk/korpusdk`.

that are possible for a single pair of words. Coordinated conjunctions depend on the first conjunct and govern the second.

6.5.11 Swedish Talbanken05 Dependency Treebank

6.5.11.1 General Characteristics

	Description
size:	∼ 321.000 tokens, 21.571 sentences
text types:	written (professional prose and students' essays
	and spoken parts (interviews and conversation debates)
dependency structure:	dependency grammar
annotation formalism:	based on Functional Generative Description (FGD) (*Sgall et al.*, 1986)
annotation formats	TIGER-XML, MALT-XML, conll

Table 6.12: Characteristics of the Swedish treebank.

6.5.11.2 Description

The Swedish dependency treebank stands out due to its division into written and spoken language. The treebank was converted from the Talbanken76 treebank that was manually annotated with a mix of dependency, constituency and topological field analysis. Punctuation can function as head and as dependent. Discontinuous constructions can occur. Coordinated conjunctions depend on the second conjunct.

6.5.12 Latin Dependency Treebank 1.4

6.5.12.1 General Characteristics

	Description
size:	∼ 30.457 tokens, 1.650 sentences
text types:	prose (classical Latin texts)
dependency structure:	dependency grammar
annotation formalism:	based on Functional Generative Description (FGD) (*Sgall et al.*, 1986)
	and on the Latin grammar of *Pinkster* (1990)
annotation formats	XML

Table 6.13: Characteristics of the Ancient Latin treebank.

6.5.12.2 Description

In our study, we also consider two ancient language treebanks - Latin and Greek. The annotation of the Latin treebank was carried out manually by three persons

– two annotators and one proof reader. Punctuation can function as head and as dependent. Discontinuous constructions can occur. Coordinated conjunctions govern both conjuncts, they can also govern the predicate if they occur at the beginning of the sentence.

6.5.13 Ancient Greek Dependency Treebank

6.5.13.1 General Characteristics

	Description
size:	$\sim$ 52.079 tokens, 3.288 sentences
text types:	prose (classical Greek texts)
dependency structure:	dependency grammar
annotation formalism:	based on Functional Generative Description (FGD) (*Sgall et al.*, 1986)
	and on the Latin grammar of *Pinkster* (1990)
annotation formats	XML

Table 6.14: Characteristics of the Ancient Greek treebank.

6.5.13.2 Description

Two different kinds of annotation were performed for Greek: the standard 3-persons-agreement annotation, as was made for Latin, and the "scholarly" 1-person annotation. The annotation scheme was adapted from those of the Latin treebank. The grammar is based on the Prague Dependency Treebank annotation (*Hajič*, 1998; *Sgall et al.*, 1986) and on the Latin grammar of *Pinkster* (1990).

6.5.14 Verbmobil Japanese Dependency Treebank

6.5.14.1 General Characteristics

	Description
size:	157.172 tokens, 17.753 sentences
text types:	spoken (appointment negotiations)
dependency structure:	HPSG based grammar converted to DG
annotation formalism:	syntactic framework taking into account the specifics
	of spoken language (repetitions, hesitations, "false starts",) *Hinrichs et al.* (2000)
annotation formats	Negra, conll

Table 6.15: Characteristics of the Japanese treebank.

6.5.14.2 Description

The Japanese treebank strongly differs from the other treebanks presented in this section. It is based on spoken language dialogs. Consequently, turns instead of sentences serve as the basic units of syntactic annotation. The treebank is transcribed in Romaji using Latin letters. Punctuation functions only as a dependent. Discontinuous constructions can occur. Coordinated conjunctions and the first conjunct are mostly governed by the second conjunct.

6.5.15 English (CoNNL) Dependency Treebank

6.5.15.1 General Characteristics

	Description
size:	∼ 993.264 tokens, 40.683 sentences
text types:	newspaper (mainly from the WSJ)
dependency structure:	dependency grammar
annotation formalism:	converted from a CPG formalism
annotation formats	conll

Table 6.16: Characteristics of the English treebank.

6.5.15.2 Description

The Treebank is a conversion from parts of the Penn treebank[19], PropBank[20] and NomBank[21] of English. Punctuation functions only as a dependent, however, other symbols such as `$ %` etc. can be heads of, for example, a number in "20%". Discontinuous constructions can occur. Coordinated conjunctions depend on the first conjunct and govern the second resulting in a chain-like structure.

6.5.16 METU Sabanci Turkish Dependency Treebank

6.5.16.1 General Characteristics

6.5.16.2 Description

The Turkish Treebank is special in the sense that it treats syntactic and morphological dependency relations as syntactic. Since Turkish is an agglutinating language, dependencies occur not only between words but also between parts of words. The treebank is compiled from the METU Turkish corpus, a balanced resource of 16 genres. Some punctuation marks can have daughter nodes, mainly when they appear with coordination and indirect speech. In addition, the main predicate is attached

[19] `http://www.cis.upenn.edu/~treebank`.
[20] `http://verbs.colorado.edu/~mpalmer/projects/ace.html`.
[21] `http://nlp.cs.nyu.edu/meyers/NomBank.html`.

	Description
size:	∼ 45.000 tokens, 5.620 sentences
text types:	novels, newspaper, etc. (16 genres of written Turkish)
dependency structure:	dependency grammar
annotation formalism:	morpho-syntactic formalism
annotation formats	XCES-XML, conll

Table 6.17: Characteristics of the Turkish treebank.

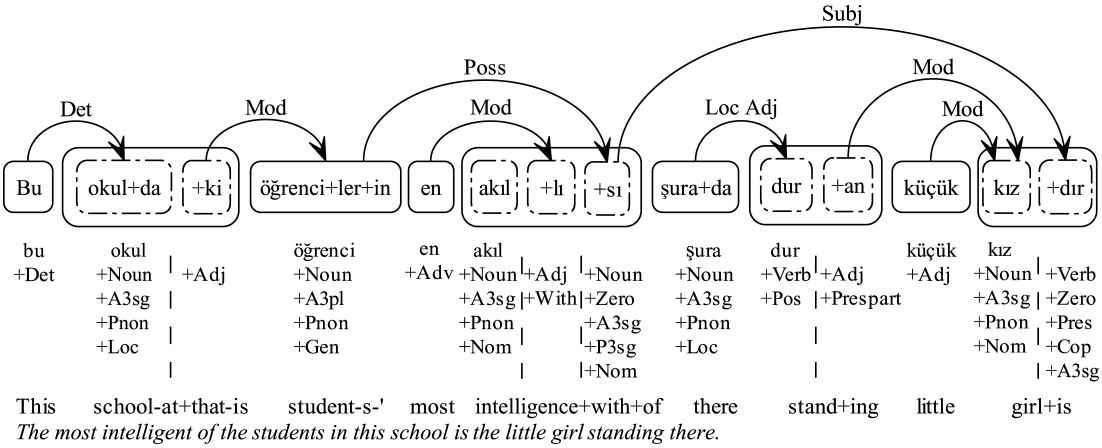

Figure 6.5: An example sentence from the Turkish treebank with its syntactic representation. Words are surrounded by triangles, IGs by dashed triangles. The dependency relations go from modifier to the head. The example is taken from (*Eryiğit et al.*, 2008, 361)

to the end-of-sentence mark (in other treebanks the end-of-sentence marks are generally attached to the virtual root). However, in the conll format (which we use here) the punctuation marks have no daughter nodes any more. This was resolved by reattaching the nodes in order to prepare the treebank for the CoNNL-X shared task representation. The annotation was performed semi-automatically with strong human supervision (*Kakkonen*, 2005).

In order to gain a better understanding of the syntactic representation of Turkish, we take a more detailed look at the peculiarities of this treebank. It is worth mentioning that Turkish nouns can expand to about 100 inflectional forms and verbs to even more forms (*Eryiğit et al.*, 2008, 361). That is, one word in Turkish can correspond to a sentence in another language, resulting in a very small average sentence length (8.6 words) reported for the Turkish treebank (*Eryiğit et al.*, 2008, 362). Dependency relations, as mentioned above, can be drawn not only between words, but rather between parts of the word. In order to account for these specifics, the authors of the treebank split the words into so called *inflectional groups* (IGs)(see Figure 6.5). In Figure 6.5 words are surrounded by solid rectangles and IGs by dashed rectangles. As can be seen in this example, the determiner word *Bu*, as well as the morpheme *ki*

can function as modifiers. Comparing this treebank to other treebanks considered so far, we can expect the number of dependency relations per sentence to be comparable in both kinds of treebank. However, the Turkish treebank has a much larger variety of candidate nodes (i.e., words and IGs) that can be selected to form a dependency relation. The last fact represents an important difference that becomes relevant in the following sections when we analyze the networks created from the treebanks described here.

6.5.17 German TIGER-DB Dependency Treebank

6.5.17.1 General Characteristics

	Description
size:	$\sim$ 700.000 tokens, 39.573 sentences
text types:	newspaper
dependency structure:	dependency grammar
annotation formalism:	converted from the TIGER treebank
annotation formats	conll

Table 6.18: Characteristics of the German treebank.

6.5.17.2 Description

The TIGER-DB treebank is a conversion from the hybrid dependency/constituency based TIGER trebank. It is based on written newspaper texts. Punctuation marks are attached to the predicate and they have no daughter nodes. Discontinuous constructions can occur. Coordinated conjunctions depend either on the first or the second conjunct.

6.5.18 Summary

In this section we summarize the quantitative characteristics of the 17 treebanks presented so far. Comparing Figures 6.6 and 6.7, we can observe how the number of tokens in the treebanks is distributed compared to the number of sentences. Czech, English and German behave similarly occupying the highest ranks of both distributions. Catalan has remarkably less sentences compared to the number of tokens (in comparison to the other languages, of course). Japanese, on the other hand, has a large number of sentences compared to a small number of words in the treebank. This fact is presumably attributed to the specifics of spoken language and the treatment of turns instead of sentences. The remaining languages' distribution is similar with respect to the number of sentences and words of particular languages.

Figure 6.8 illustrates the number of tokens (sorted in descending order) compared to the number of types in each treebank. The *type-token ratio* (TTR) ($TTR = (\frac{types}{token}) * 100$) is an index applied in quantitative linguistics for measuring the lexical richness

Language	token #	types #	TTR in %	sentences #	genres #
BUL	196.000	32.421	16.5	12.823	3
CZE	1.290.000	146.504	11.3	88.374	4
RUS	256.800	58.373	22.7	17.628	>4
SLV	30.000	8.343	27.8	1.998	1
DAN	100.000	19.133	19.1	5.512	>4
DUT	172.000	28.475	16.5	13.349	1
ENG	993.264	44.748	4.5	40.683	1
GER	700.000	72.630	1.0	39.573	1
SWE	321.000	25.097	7.8	21.571	4
JAP	157.172	3.271	2.1	17.753	1
CAT	478.000	38.882	8.1	16.631	1
ITA	41.544	7.986	19.2	1.500	2
LAT	30.457	8.326	27.3	1.650	1
ROM	36.150	8.867	24.5	4.042	1
SPA	125.000	17.101	13.68	3.512	4
TUR	45.000	19.386	43.1	5.620	>4
GRC	52.079	11.521	22.1	3.288	1

Table 6.19: The table lists the number of tokens, types, the type-token ratio (TTR), sentences and genres for 17 treebanks sorted by language families.

of a text. The TTR depends on the number of tokens, thus, direct comparisons of treebanks according to this measure do not make much sense (because the treebanks are of varying size). Higher number of tokens results in a lower TTR, thus, only treebanks of roughly equal size can be compared and only with caution since factors such as the number of genres in the sample, the morphological variety of the language, etc. may bias the coefficient. In order to nevertheless get a picture of the distribution of types versus token in our data, we plotted the languages sorted by the number of token in a descending order, and the number of types on a log-to-log plot (Figure 6.8). When comparing only languages in the local neighborhood (i.e., of the same number of token), we could observe that DUT and BUL nicely run parallel according to the number of tokens and the number of types (both have the TTR: 16.5). In contrast, despite having a similar number of tokens, DUT and JAP differ significantly (TTR: 16.5 vs. 2.0). JAP is in fact an outlier featuring an extremely low number of types in relation to the number of tokens. Again, the reason may be that the written genre features richer vocabulary than the spoken one (*Williamson*, 2009). Another outlier is TUR, exhibiting an extremely high number of types (TTR: 43.0) compared to, for example, ITA (TTR: 19.2) which shares nearly the same number of token. The last obeservation results presumably from the morpho-syntactic dependency annotation. In the agglutinating language every single morpheme can function as a separate type raising the overall number of types in the treebank.

6.6 Constructing Global Syntactic Dependency Networks

For every language we extracted a Global Syntactic Dependency Network (GSDN) as introduced by *Ferrer i Cancho et al.* (2004) and compared languages according to these networks. The following section describes the treebanks and the extraction procedure.

6.6.1 Network Definition

The notion of GSDN goes back to (*Ferrer i Cancho et al.* 2004) who defined a GSDN as "a set of n words $V = \{s_i\}(i = 1, ..., n)$ and an adjacency matrix $A = \{a_{ij}\}$. If a link connects the modifier s_i with the head s_j then $a_{ij} = 1$ (and $a_{ij} = 0$ otherwise)." In this case, links go from the modifier to the head, of course, this can be changed the other way round. According to this definition, GSDNs are *simple directed graphs*, however, complex network theoretical measures applied to characterize GSDNs in this thesis treat them as undirected.

We use word forms or *types* as vertices of the network, since not all treebanks are lemmatized. Two vertices (i.e., types) of a GSDN are linked if they appear at least once in a modifier-head relation in the treebank.

6.6.2 From a Dependency Treebank to GSDN

The procedure of creating a GSDN is illustrated in Figure 6.9. The treebank is parsed sentence by sentence and new words are added to the network. Words are linked according to the dependency relations they constitute. When a word is already present in the network (e.g. *book* in Figure 6.9), more links are added to it. Finally, we get a network containing all words and all dependency relations of a particular treebank. The degree of a word gives the number of different dependency relations with other words.[22] As mentioned elsewhere in the previous sections, punctuation marks and special symbols were not included in the network in order to get the same representation of word-to-word dependency relations for every language.

6.7 Summary

In this chapter we illustrated the data we will be using in the subsequent experiments. We presented 17 dependency treebanks and their qualitative and quantitative characteristics. Issues on the specifics of dependency theories were discussed especially with respect to our data. In order to account for a nearly optimal comparability of data, we tried to eliminate the differences among the treebanks resulting from theory and language specific deviations. A unification of representation formats was also performed. Finally, the procedure of creating a global syntactic dependency network (GSDN) was described. After having created a GSDN for each treebank, we applied a range of network characteristics to the networks in order to learn more about the

[22]Note that weights of edges are not considered by this model, that is, if two words occur more than once in a modifier-head relation, it does not result in an increase of degrees of these words.

respective languages looking at the global typology of their networks. The network characteristics used are presented in the following chapter.

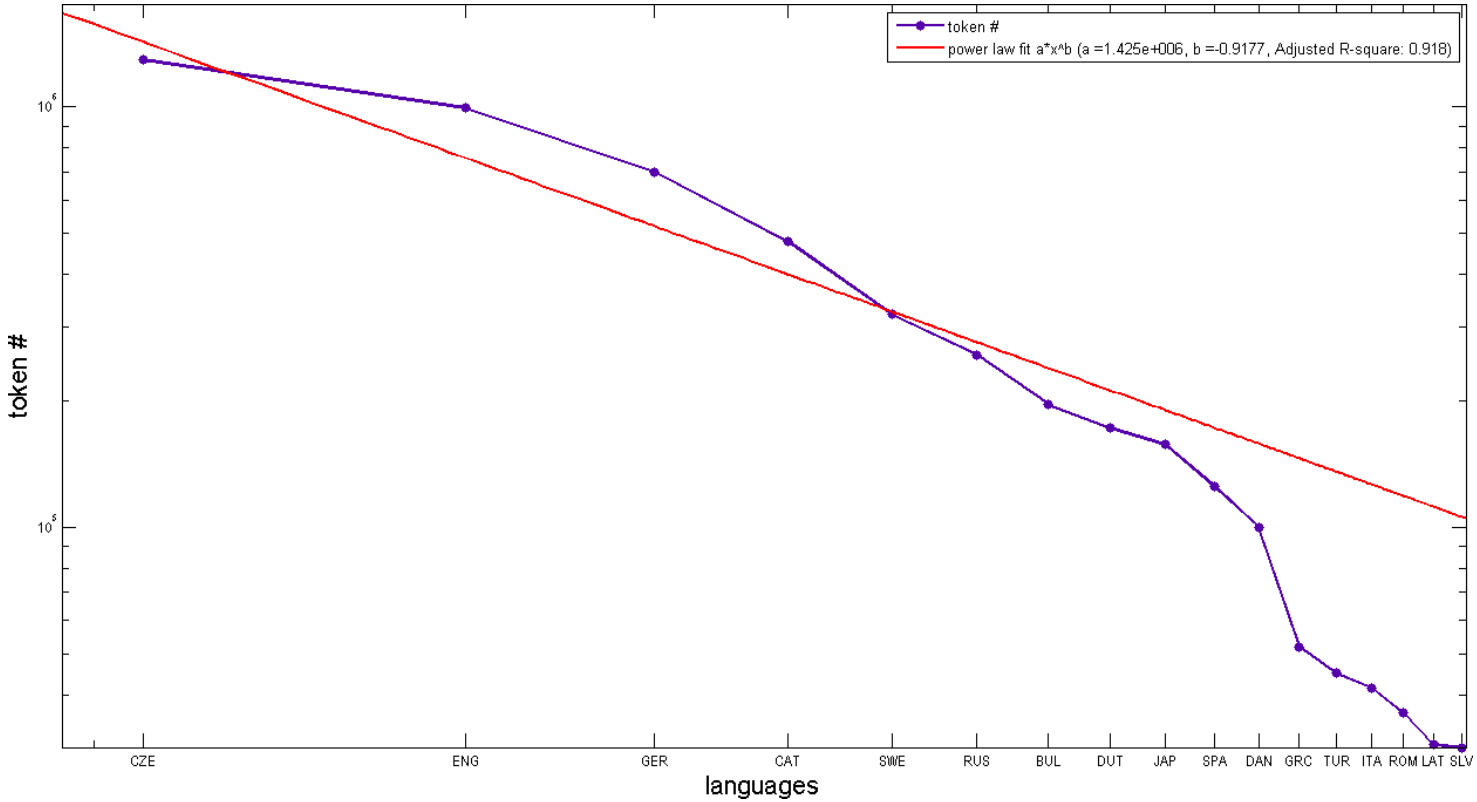

Figure 6.6: The distribution of 17 treebanks sorted by the number of tokens in a descending order on a log-log plot. The treebanks on the x-axis are abbreviated with their language codes. The distribution follows a power law with a negative decay with a certainty of 93 % according to the adjusted coefficient of determination.

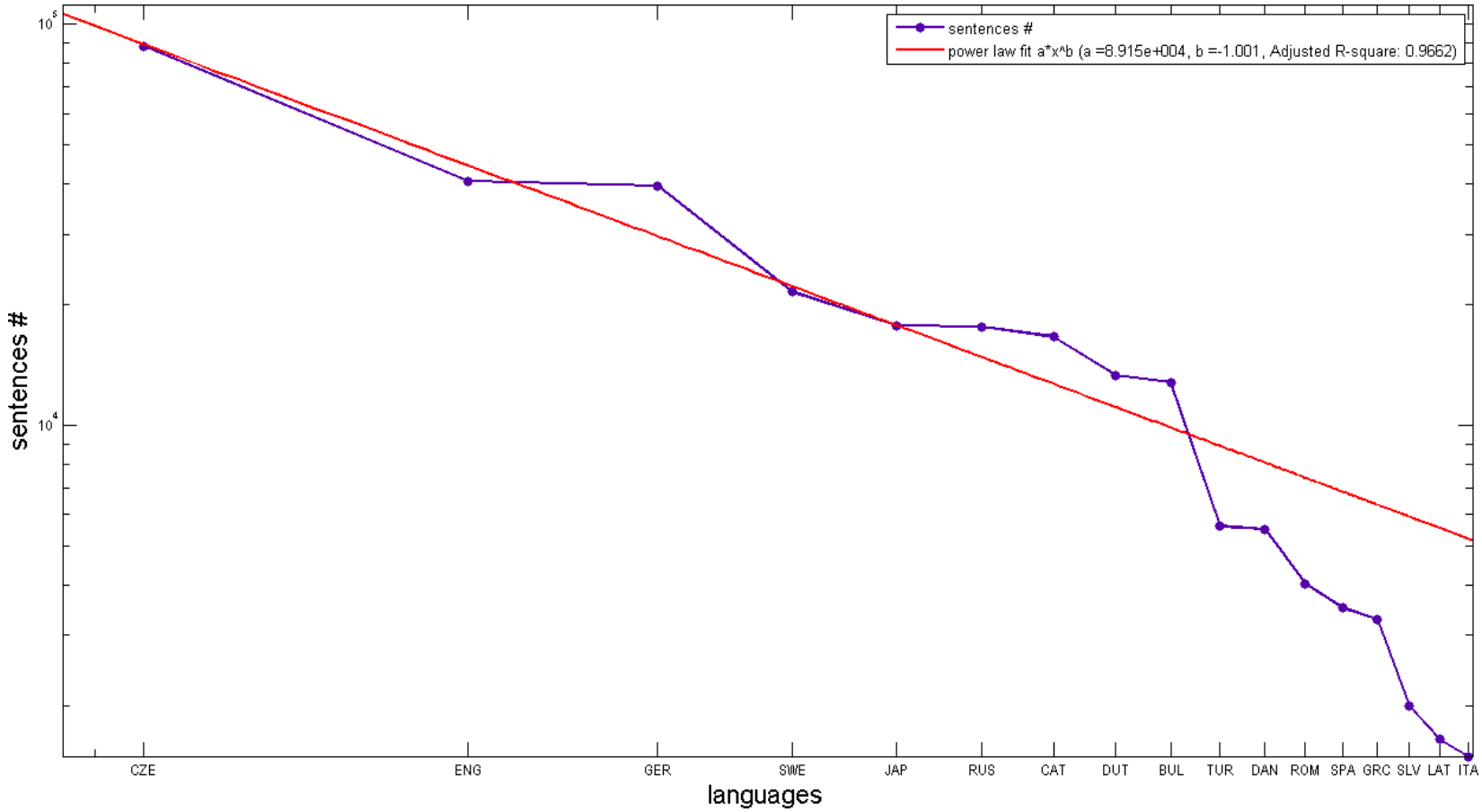

Figure 6.7: The distribution of 17 treebanks sorted by the number of sentences in a descending order on a log-log plot. The treebanks on the x-axis are abbreviated with their language codes. The distribution follows a power law with a negative decay with a certainty of 98 % according to the adjusted coefficient of determination.

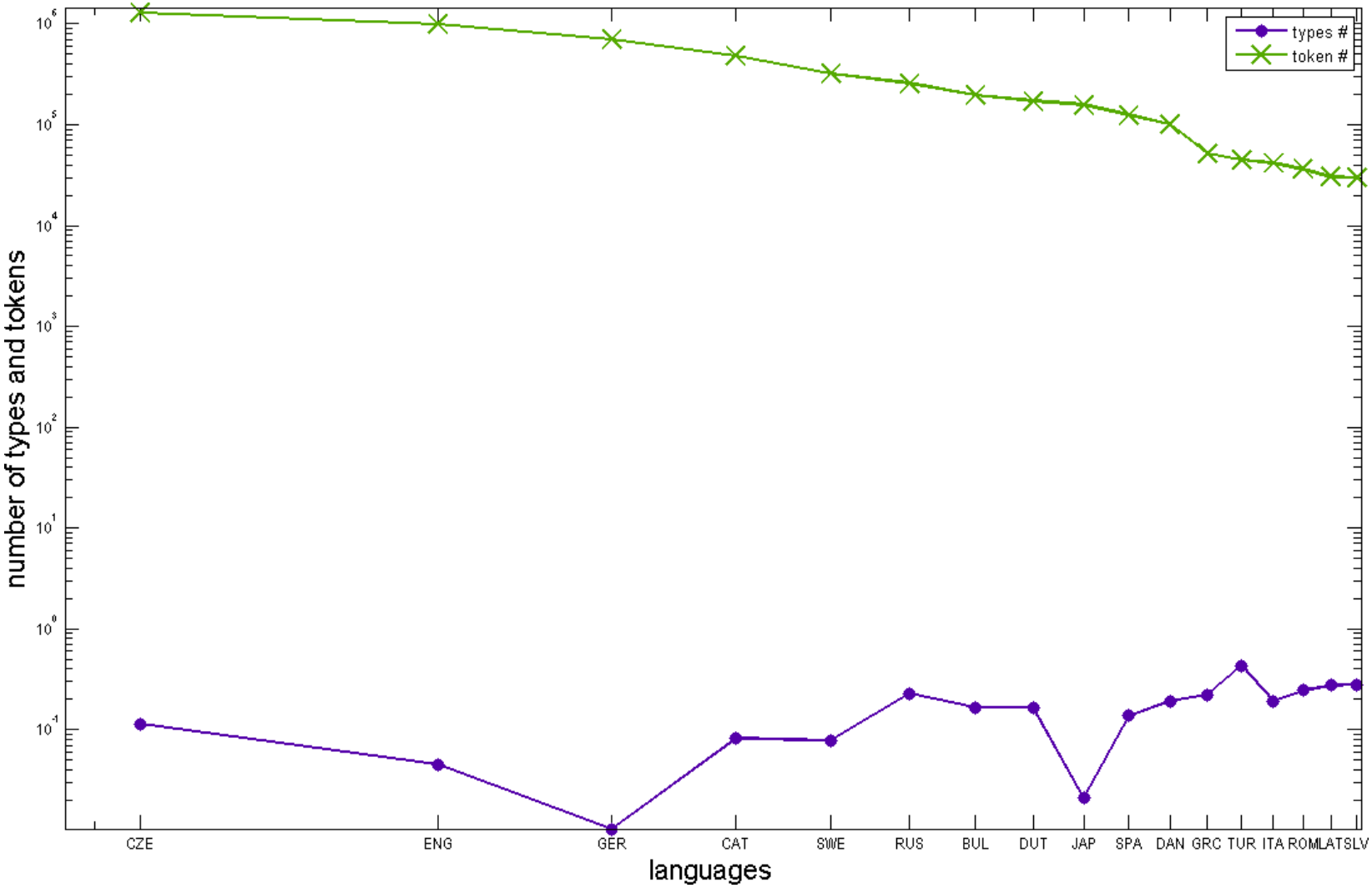

Figure 6.8: The distribution of 17 treebanks sorted by the number of tokens in a descending order on a log-log plot. The treebanks on the x-axis are abbreviated with their language codes.

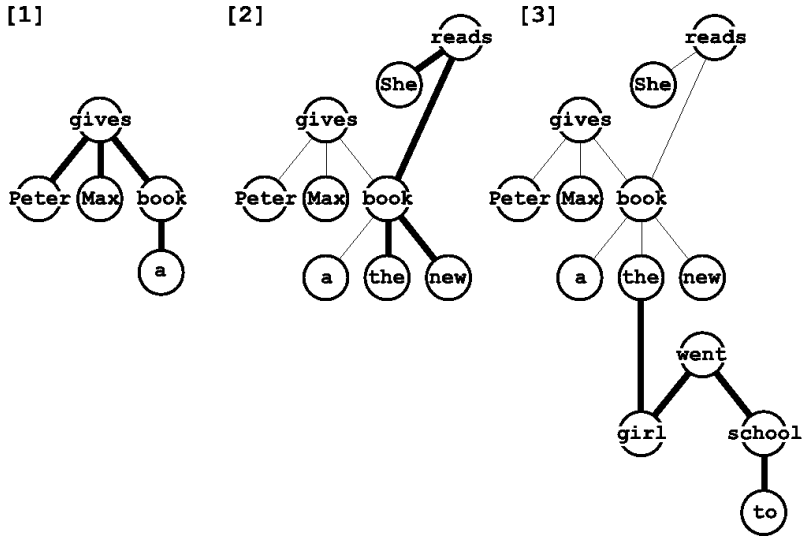

Figure 6.9: The figure taken from (*Mehler et al.*, 2010a) exemplifies how a GSDN is created after parsing the $1, 2, 3$ sentences.

CHAPTER VII

Network Indices

7.1 Introduction

Index	Feature	Short Description	Area
F_1	$C_{ws}(G)$	the cluster coefficient of G	1
F_2	$C_{br}(G)$	the cluster coefficient of G	1
F_3	$L(G)$	the average geodesic distance of G	1
F_4	$D(G)$	the diameter of G	1
F_5	$r(G)$	the degree of assortative mixing of G	1
F_6	$\epsilon(G)$	the average degree of G	1
F_7	$\mathsf{lcc}(G)$	the fraction of the largest connected component of G	1
F_8	$\gamma(G)$	the γ of the power law of type $Ck^{-\gamma}$ which best fits to the degree distribution of G	1
F_9	$\bar{R}^2_{\gamma}(G)$	the corresponding adjusted coefficient of determination	1
F_{10}	$\gamma_S(G)$	the γ of $Cn^{-\gamma}$ which best fits to the size distribution of connected components of G	1
F_{11}	$\bar{R}^2_{\gamma_S}(G)$	the corresponding adjusted coefficient of determination	1
F_{12}	$\gamma_{\bar{k}_{nn}(k)}(G)$	the γ of the power law of type $Ck^{-\gamma}$ which best fits to the distribution of $\bar{k}_{nn}$ values of G	1
F_{13}	$\bar{R}^2_{\gamma_{\bar{k}_{nn}(k)}}(G)$	the corresponding adjusted coefficient of determination	1
F_{14}	$\gamma_{C(k)}(G)$	the γ of the power law of type $Ck^{-\gamma}$ which best fits to the distribution of $C(k)$ values of G	1
F_{15}	$\bar{R}^2_{\gamma_{C(k)}}(G)$	the corresponding adjusted coefficient of determination	1
F_{16}	$\mathsf{GC}(G)$	the graph centrality of G	2
F_{17}	$\mathsf{CC}(G)$	the standard deviation of the closeness centrality of G	2
F_{18}	$\mathsf{DC}(G)$	the degree centrality of G	2
F_{19}	$\mathsf{Cp}(G)$	the compactness of G	3
F_{20}	$\mathsf{Ch}(G)$	the cohesion of G	3
F_{21}	C_A	the relative graph connectivity (*Mehler et al.*, In preparation)	3

Table 7.1: The table lists composite features from the model of *Mehler* (2008a) applied in the present study. They fall into three groups, which are features of *complex network theory* (1), *social network analysis* (2) or *hypertext structure analysis* (3) (as indicated in the last column).

In this chapter we look more closely at the network indices used in order to characterize the GSDNs. We refer here to features summarized in Table 7.1, which were calculated by Alexander Mehler (see *Abramov and Mehler* (2011)) and served as input to the clustering algorithm. Before discussing the coefficients, here are some

basic definitions we operate with. A graph $G = (V, E)$ is a GSDN (see Section 6.6). A degree $d(v_i)$ of a vertex v_i is the number of edges directly incident to v_i, so that $d(v_i) = |E(v_i)|$ (*Diestel*, 2006).

7.2 Indices - Description, Definition, Interpretation

In the following, we discuss single network indices, provide their definitions and possible interpretations with respect to GSDNs.

7.2.1 Average Geodesic Distance

7.2.1.1 Description

In many large scale networks (natural or random) the average distance between two vertices taken at random is small compared to the size of the network, so "it scales logarithmically or slower with the number of vertices" (*Caldarelli and Vespignani*, 2007). This property, is also called the *small-world effect* (*Caldarelli and Vespignani*, 2007), which means for example in a social network that two vertices (persons) selected at random are separated from each other by a small number of steps (six-degrees of separation). Random graphs à la *Erdős and Rényi* have this property too.

7.2.1.2 Definition

The average geodesic distance L of a graph G is calculated as the average of the shortest paths between each pair of vertices in G.

7.2.1.3 Interpretation

In the case of language networks we can expect distances to be short in general since, for example, content words or nouns are linked to function words whose number (in natural language) is of limited size. This fact assures the occurrence of short paths in a GSDN. Thus, the fact that a language utilizes grammatical and lexical words (morphemes) is responsible for the small-world effect in this kind of linguistic network. If a language consisted of an infinite number of lexical morphemes, L would increase at a higher rate together with the size of the network. Humans do not posses an unlimited memory capacity, and this kind of network accounts for both - sparing of resources and fast information transmission (*Ferrer i Cancho*, 2003; *Mehler*, 2008a). Small-world effects are, thus, indispensable for GSDNs.

Differences among languages, however, are expected resulting in longer paths for morphologically richer languages like Russian than for analytic languages. The reason is that analytic languages have more grammatical morphemes (like, e.g., prepositions) that connect to various different word forms. This kind of vertices serve as short cuts reducing the paths in the network. When we look at Table 15, Russian and Czech have higher L's than Swedish and Danish. So are languages such as Swedish more efficient

in terms of information storage and transmission than, for example, Russian? On the one hand, on the syntactic-dependency level this may seem true. But on the other hand, much information about the sentence structure is coded within the word (in a synthetic language) and decoded by applying a limited number of word formation rules learned and stored in the memory. As shown in Chapter IV - much more structural information can be deduced from Russian words, than from English. This means, there is a mutual connection between the complexity on the morphological and syntactic levels as predicted by the synergetic model of (*Köhler*, 1986). Reducing the complexity on the morphological levels results in an increase on the syntactic level and so on. Indices such as L are able to measure this complexity on a particular level allowing us to get an automatic characteristic of a language.

7.2.2 Average Degree

7.2.2.1 Description

The average degree $\epsilon(G)$ of a graph G (F6 in Tab. 7.1) is in principal a very informative feature representing the proportion of edges with respect to the number of vertices.

7.2.2.2 Definition3

The average degree is defined as follows:

$$\epsilon(G) = \frac{edges}{vertices} \tag{7.1}$$

7.2.2.3 Interpretation

Similar to average geodesic distance, we can expect an analytic language to have more edges (and proportionally fewer vertices) since the same morphological forms are used more frequently. Thus, the average degree of an *analytic* graph should be higher than the average degree of a *synthetic* graph.

Like in the case of $L(G)$, this expectation is confirmed by high values of ϵ for Catalan, Swedish and English ($\sim$ 5) in comparison to languages such as Slovene, Russian and Bulgarian (see Table 7.2).

7.2.3 Clustering

7.2.3.1 Description

The degree of clustering or *transitiviy* in a network is attributed to the "tendency of a network to form cliques in the neighborhood of any given vertex."[1] In a highly clustered network the occurence of cliques is very likely. If a vertex a is connected to b and b to c, then a connection between a and c is likely to exist. High network clustering together with short average geodesic distance L constitute the *small-world model* of

[1]See *Caldarelli and Vespignani* (2007, 12).

Watts and Strogatz (1998), which is applicable to many real world networks (e.g., social, biological, linguistic, etc.). This goes together with an interesting observation that random graphs exhibit short L but small clustering, whereby regular grids are highly clustered but have long L. Small-world networks exhibit both short L and high clustering.[2] This effect was also confirmed for some linguistic networks[3] and for GSDNs in particular[4]. In this chapter, we take a closer look at these properties with respect to GSDNs.

7.2.3.2 Definition

Given a vertex i with a degree k_i, we denote e_i as the number of edges between the k_i neighbors of i. Then, "the clustering coefficient c_i is defined as the ratio between the actual number of edges e_i, and the maximally possible e_i among the neighbors of i" (*Caldarelli and Vespignani*, 2007, 12):

$$c_i = \frac{e_i}{k_i(k_i-1)/2}. \tag{7.2}$$

The clustering coefficient can thus be interpreted as the average probability of connectivity among the neighbors of a vertex. The clustering coefficient is $c_i \equiv 0$ for all $k_i \leq 1$.

In order to characterize the overall clustering of a graph, we compute two different clustering coefficients: $C_{ws}(G)$ (*Watts and Strogatz*, 1998) and $C_{br}(G)$ (*Bollobás and Riordan*, 2003) (see Features F1 and F2 in Tab. 7.1). $C_{ws}(G)$ is simply the average of all c_i's in the graph (*Barrat et al.*, 2008, 11):

$$C_{ws} = \frac{1}{N}\sum_i c_i. \tag{7.3}$$

Another possibility to compute the graph related clustering value is to weight the number of transitive relations in the graph by the degrees of the vertices. This is done by the following coefficient (see *Bollobás and Riordan* (2003)):

$$C_{br} = \frac{\sum_i (k_i(k_i-1)/2)c_i}{\sum_i k_i(k_i-1)/2} = \frac{\sum_i e_i}{\sum_i k_i(k_i-1)/2}. \tag{7.4}$$

This coefficient considers the sum of connected neighbors of c_i in relation to the number of all possible transitive connections in the graph. C_{br} is presumably more precise in characterizing networks, since not simply the average clustering is considered but the actual transitivity in relation to the maximally possibly transitivity of the particular graph (i.e., according to its vertex degrees).

While the two different coefficients result in "different values of clustering for a given graph" (*Barrat et al.*, 2008, 11), we analyze both of them in GSDNs and compare the results in the following sections. In the next section we relate the concept of clustering to GSDNs in order to understand the role of triangles for characterizing syntactic networks.

[2] See *Caldarelli and Vespignani* (2007).

[3] See e.g. *Mehler* (2008a,b); *Mehler et al.* (2010a)

[4] *Ferrer i Cancho et al.* (2004, 2007); *Abramov and Mehler* (2011).

Figure 7.1: Two Example Sentences in Dependency Notation.

7.2.3.3 Interpretation:

In our case we deal with graphs consisting, for instance, of verbs linked to nouns (see Fig. 7.1), nouns linked to articles, adjectives, etc. Edges occur mostly among different word forms: *verbs-nouns*, *nouns-adjectives*, etc. This means the probability of triangle relations reaching, for example, from *Peter* to *gave*, from *gave* to *book* and back from *book* to *Peter* is very low (Fig. 7.1). This, in turn, results in a low clustering coefficient for dependency networks in general. Due to dependency syntax, nouns linked to nouns or verbs to verbs should not occur in simple sentences. However, in the case of sentences like "I know Peter read the book" (Fig. 7.1) "know" and "read" are linked, and since in another sentence "know" and "Peter" may be linked too, the three words "know", "read" and "Peter" will form a triangle. The above example explains how triangles can appear in language networks. That means, we can expect triangles to be present to some extent in all languages.

The interesting question in this context is whether we can distinguish languages based on the amount of triangles, that is, on the value of the clustering coefficient. Languages such as, for example, Swedish are more analytic than Russian; thus Russian has more individual word forms representing different inflectional cases than Swedish. When we transfer this observation into networks we can expect Russian to have a lower clustering value than Swedish. In the example above the word "Peter", for instance, would be written differently depending on the inflectional case (e.g. nominative vs. accusative) which minimizes the probability of a triangle containing, for instance, "know" and "read". For an analytic language such a connection is more probable due to sparse morphological variation. Thus, a relation like in the above example could frequently occur in English, Swedish, etc. but not as frequently in Russian.

We expect higher clustering for Slavic, rather than for Germanic languages.

7.2.4 Degree Distribution

7.2.4.1 Description

One of the most basic statistical characteristics of graphs is the distribution k_i of vertex degrees. For an undirected graph the degree distribution $P(k)$ is the prob-

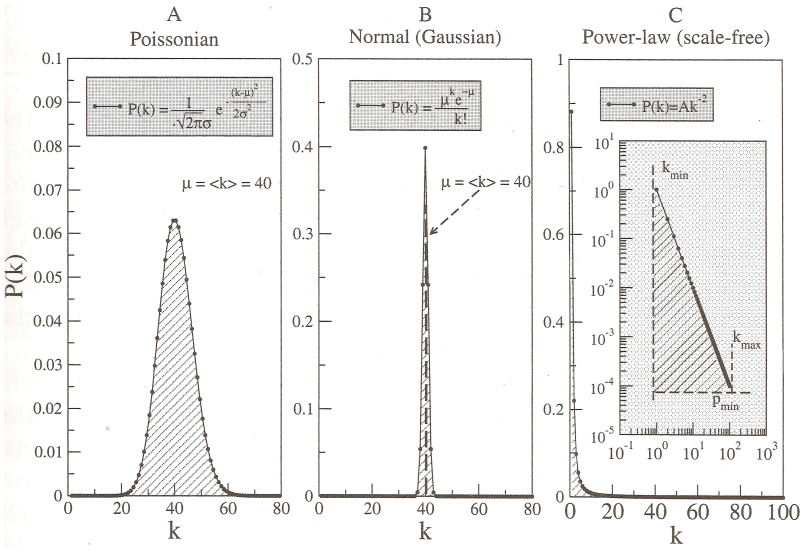

Figure 7.2: The plot is taken from *Caldarelli and Vespignani* (2007, 13), it shows (A) the Gaussian, (B) Poisson and (C) Power-law distributions.

ability of a randomly taken vertex i to have the degree k.[5] The distribution can be functionally described by two classes of networks - *homogenous* and *heterogenous* networks. Figure 7.2, taken from *Caldarelli and Vespignani* (2007), illustrates these functions. Homogenous networks have typically the form of the Poissonian (A) or Gaussian (B) distributions. These networks are easily characterized by the average degree and standard deviation. The average degree represents a typical value of a homogeneous network, a value that a randomly chosen vertex will take with a high probability. In heterogenous networks (power-law), in turn, typical degrees are small degrees due to the long tail. However, all other degrees (large and intermediate) are also probable, so that the average degree loses its expressiveness in this kind of networks. "In case of distributions with a power-law tail with exponent $2 \leq \gamma \leq 3$ we have that the fluctuations are unbounded and depend only on the system size."[6] The networks are also called *scale-free* networks since there is no characteristic scale to describe them. This property can also "be extended to values of $\gamma \leq 2$ [...]."[7]

7.2.4.2 Definition

We consider the probability distribution $P(k)$, and check the availability of the long tail. If

$$P(k) \sim k^{-\gamma}$$

holds, we deal with heterogenous (scale-free) networks. We check whether our networks fit this model by looking at the exponent $\gamma(G)$ and the adjusted coefficient of

[5] See *Caldarelli and Vespignani* (2007, 12).
[6] *Caldarelli and Vespignani* (2007, 14).
[7] Ebd.

determination $\bar{R}^2_\gamma(G)$ which evaluates the goodness of the fit (Features F8 and F9 in Tab. 7.1). The advantage of γ is that it allows to characterize the network independent of its size (i.e., γ describes the slope of the distribution). This is an important factor when we deal with GSDNs, which all have varying sizes and orders.

7.2.4.3 Interpretation:

Degree distribution in GSDNs is directly related to Zipf's law of word frequencies (*Zipf*, 1932). Thus, we can study degree distributions in analogy to word frequency distributions (right-skewed, long-tail); Zipf's law can be directly applied to the degrees of our language networks.

As predicted by Zipf's law and shown in (*Ferrer i Cancho et al.*, 2004, 2007), all values of $\gamma(G)$ for GSDNs are negative, and $\bar{R}^2_\gamma(G)$ is close to 1. Thus, GSDNs are heterogenous networks. However, the values of these features can vary among languages reflecting different frequencies of dependency relations that form the shape of the distribution. Why? As in the case of airport or WWW networks (see *Barrat et al.* (2008)) in all languages we can expect *hubs*, that is, vertices (words) of a high degree. Analytic languages should have more hubs in general, since more high-frequent and easily attachable grammatical morphemes are used to form a sentence. These hubs can have nearly the same degrees, which would flatten the distribution from highly skewed to more even, resulting in higher values of γ for analytic languages than for synthetic. Synthetic languages, in turn, which express grammatical relations within the word, should feature to a lesser extent connected hubs, more skewed degree distributions and smaller values of γ respectively. As we will see later in the Chapter, the above argumentation is confirmed by the results.

7.2.5 Connectivity Correlations

7.2.5.1 Description

Connectivity correlation or *assortativity* is a property of a network that describes connectivity preferences among its vertices. The question here is whether vertices of degree k connect to vertices of similar degrees (assortative mixing) or not (disassortative mixing). Assortativity is thus an increasing or decreasing function of k. Note that scale-free and highly clustered networks can be either assortative or disassortative, so assortativity serves as an additional index to characterize the network. Assortative mixing was observed, e.g., for social networks (*Newman*, 2003) such as co-authorship, company director networks, etc. Disassortative mixing was shown for biological (e.g., protein interaction, metabolic networks, food webs, etc.), technical (*Pastor-Satorras et al.*, 2001) such as WWW-links, Wiki and document networks (*Mehler*, 2008a).[8] All the 6 GSDNs analyzed in *Ferrer i Cancho et al.* (2007) exhibit disassortative mixing. In the present study we can confirm the above finding for 17 GSDNs.

Different means to measure connectivity correlations of networks were proposed in the literature. We focus on three widely used coefficients: The *correlation coefficient*

[8]See *Caldarelli and Vespignani* (2007) for review.

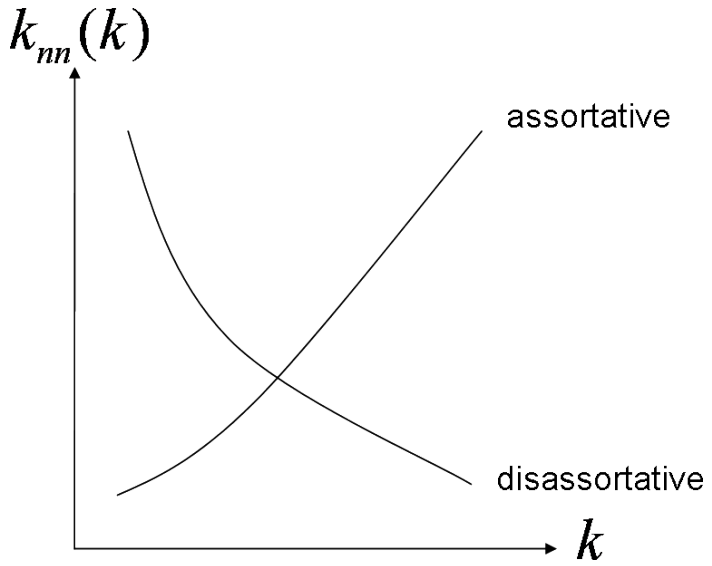

Figure 7.3: Assortative vs. disassorrtative mixing *Barrat et al.* (2008, 15).

of (*Newman and Park*, 2003) (feature $F5$ in Table 7.1) the *connectivity correlation* (*Pastor-Satorras et al.*, 2001) (features $F12$ and $F13$ in Table 7.1), and two combined features based on the clustering coefficient (features $F14$ and $F15$ in Table 7.1).

7.2.5.2 Definitions:

- The Pearson correlation coefficient of (*Newmann*, 2002) can be used to compute the assortativity value $r(G)$ of a network considering all possible degrees (see *Mehler* (2008a)). The Pearson correlation coefficient is defined as follows:

$$r(G) = \frac{\sum_e j_e k_e / E - \left[\sum_e (j_e + k_e)/(2E)\right]^2}{\left[\sum_e (j_e^2 + k_e^2)/(2E)\right] - \left[\sum_e (j_e + k_e)/(2E)\right]^2}, \tag{7.5}$$

  with j_e and k_e being the degrees of the extremities of edge e and E the total number of edges. A positive value of $r(G)$ indicates assortative mixing of the graph, a negative value the opposite (disassortativity), zero indicates no correlation (*Barrat et al.*, 2008).

- Another index of connectivity correlation $\bar{k}_{nn}(k)$ is computed as the average degree of the nearest neighbors of vertices with degree k. This measure results from an average of nearest neighbors degrees of a vertex i:

$$k_{nn}(i) = \frac{1}{k_i} \sum_{j \in V(i)} k_j, \tag{7.6}$$

  with $V(i)$ being the set of the nearest neighbors of i. The degree correlation function $\bar{k}_{nn}(k)$ is then computed as follows:

$$\bar{k}_{nn}(k) = \frac{1}{N_k} \sum_{i=1}^{N_k} k_{nn}(i), \tag{7.7}$$

where N_k is the set of vertices with degree k.[9]

The value of $\bar{k}_{nn}(k)$ increases when the network exhibits assortative mixing, and decreases when the network features disassortative mixing (see Fig. 7.3 taken from *Barrat et al.* (2008, 15) for illustration). If no correlations occur, $\bar{k}_{nn}(k)$ is a constant. $\bar{k}_{nn}(k)$ is a better indicator of assortativity than $r(G)$ since the complete degrees' distribution of the nearest neighbors is considered.

- Yet another measure of connectivity correlation is the distribution of clustering coefficients $\bar{c}(k)$[10], which constitutes the probability of clustering dependent on the degree k. Due to the functional dependence of local vertex clustering on the degree, $\bar{c}(k)$ behaves power-law-like

$$\bar{c}(k) \sim k^{-\alpha}$$

as observed for many scale-free networks (*Ravasz and Barabási*, 2003),[11] especially for "uncontrolled, self-evolving" networks (e.g., WWW, biological, linguistic, social, etc.) as emphasized by *Krioukov et al.* (2004). More controlled, technical networks (such as the internet at router level or the power grid of Western US) do not exhibit a power-law decay of clustering coefficients (*Ravasz and Barabási*, 2003). Since GSDNs belong to the category of linguistic networks, they are assumed to behave alike the real-world networks examined so far. We compare the values of $\gamma_{\bar{c}(k)}$ and the adjusted coefficient of determination $\bar{R}^2_{\gamma_{\bar{c}(k)}}$ for GSDNs to check this.

7.2.5.3 Interpretation:

Assume, for example, that nouns and verbs have the same degrees. Then, we can ask whether they are connected to each other or not. If they are, it indicates assortative, if not disassortative mixing. As was shown in *Ferrer i Cancho et al.* (2007), and also confirmed here, all GSDNs exhibit disassortative mixing. A more interesting question in our context is whether assortativity allows to separate genealogically different languages. In general, we expect all the combined features (i.e., both γ's and $\bar{R}^2$'s, and $r(G)$) to correlate to some extent.

We assume the features $\gamma_{\bar{c}(k)}$ and $\bar{R}^2_{\gamma_{\bar{c}(k)}}$ to be biased by the size of the network. Since we deal with networks of varying size, the accordance on the power-law distribution might be worse for small, and better for large networks. This is because for vertices with a degree ~ 2 the difference between actually connected, and maximally possible connected neighbors (see c_i) is small. If we do not take the degree of the vertex into account (as in $c_{br}(i)$), we get high cluster values for low degree vertices even if there is only one triangle rooted by such a vertex. The clustering coefficient C_{ws} averages over all values, and such biases become less valuable than if we consider the distribution $\bar{c}(k)$. In case of small size networks, $\bar{c}(k)$ can be influenced by many

[9] See *Barrat et al.* (2008) for details.

[10] We compute the distribution of $c_{ws}(i)$ values here.

[11] See (*Barrat et al.*, 2008).

low degree vertices with high clustering. Thus, this measure may prove inappropriate when comparing networks of different sizes.[12]

The same factor may influence the distribution of $k_{nn}(k)$ too, though to the lesser extent. We expect $k_{nn}(k)$ to behave much alike the degree distribution. The goodness of fit expressed by $\bar{R}^2_{\gamma_{k_{nn}(k)}}$ may by biased by the size, and is presumably not that informative in the typological sense. $\gamma_{k_{nn}(k)}$ in turn, may prove a very informative typological feature. Here, we expect the syntactic connectivity preferences to come into the fore. Maybe this feature is not that interesting for classification but it might reflect the individual properties of a particular language. $\gamma_{k_{nn}(k)}$ describes the slope of the degree spectrum of nearest neighbors of all vertices with the degree k. That is, transitive relations between vertices of particular degrees are inspected. On the one hand, looking at vertices on a particular rank we might learn which word forms connect to which other in a particular language. On the other hand, the specifics of the syntactic theory might influence the resulting connectivity preferences. We will discuss this in more detail in this chapter's results section.

Finally, we focus on the correlation coefficient $r(G)$ of *Newmann* (2002). *Barrat et al.* (2008) state that "such a measure can be misleading when a complicated behavior of the correlation functions (non-monotonous behavior) is observed. In this case the Pearson coefficient gives a larger weight to the more abundant degree classes, which in many cases might not express the variations of the correlation function behavior."[13] Such a non-monotonous behavior is also highly probable for our GSDNs, especially for those of them with small size. This becomes evident from not $\bar{R}^2_{\gamma_{k_{nn}(k)}}$ and $\bar{R}^2_{\gamma_{\bar{c}(k)}}$ not being congruent (about 0.8 or 0.7 respectively). Typologically, this makes the feature rather problematic, which however must not necessarily diminish its contribution to the result of the classification.

7.2.6 Centrality

7.2.6.1 Description

We use various centrality measures: the *degree centrality* (DC) (*Feldman and Sanger*, 2007), *graph centrality* (GC) (*Hage and Harary*, 1995) and the *(standardized) closeness centrality* (CC) (*Wasserman and Faust*, 1999). All indices rank the GSDNs within the interval $[0, 1]$ with 1 indicating high, and 0 low overall centrality. Centrality is calculated for single vertices and then aggregated by means of an aggregation function in order to achieve a single characteristic value for a graph.[14]

An interesting question concerning GSDNs and centrality is the following: *Are there words that can have a dependency relation with almost every other word?* It is rather unrealistic to assume only one such an universally attachable word to appear in a language (in that case the centrality of a graph would be equal to 1). The idea

[12] A better solution would be to look at the distribution of $c_{br}(i)$, which is presumably more informative.

[13] See *Barrat et al.* (2008, 14).

[14] Note, that all indices are computed only for the **L**argest **C**onnected **C**omponent (*lcc*). This is, of course, an abstraction and some information might be lost.

that all words are equally probable to connect to all other words in a GSDN is also unrealistic. This would also contradict the Zipfian distribution of words in language. Thus, realistic centrality values of languages lie within a particular interval. The question is how large the variation in this interval is? And does this variation tell us anything about the typological properties of a particular language? Keeping these questions in mind, in this section, we look more closely at the centrality values of GSDNs exemplified by degree centrality. We have selected these measures since they represent two classes of centrality measures - degree based (i.e., DC) and distance based (i.e., GC, CC).

7.2.6.2 Definitions:

- *Degree centrality* (DC): relates vertex degrees to each other. If there are many vertices of a low degree and one vertex of a high degree connected to all the others, the centrality of the graph will be high (DC ~ 1).[15] DC is defined as follows:

$$\mathsf{DC}(G) = \frac{\sum_{v \in V} d_{\max}(V) - d(v)}{(|V|-1)(|V|-2)} \in [0,1] \tag{7.8}$$

  In the above equation each vertex is compared to $d_{\max}(V)$, that is, the vertex with the highest degree in the graph. The fewer the vertices equal to $d_{\max}(V)$, the higher is the DC. The DC value of 1 corresponds to a graph of a form like a *star graph* with one vertex of the maximal degree. A graph has a DC value of 0 if each vertex has the same degree.

- *(Standardized) Closeness Centrality* $\mathsf{CC}(v)$ (*Wasserman and Faust*, 1999) is another vertex related index of centrality which is a function of geodesic distances rather than the degree of a vertex (*Feldman and Sanger*, 2007). The $\mathsf{CC}(v)$ is defined there as follows:

$$\mathsf{CC}(\mathsf{v}) = \frac{|V|-1}{\sum_{w \in V} gd(v,w)} \in [0,1] \tag{7.9}$$

  The closeness centrality of a graph $\mathsf{CC}(G)$ is computed by *Mehler* (2008a) as follows:

$$\mathsf{CC}(\mathsf{G}) = \begin{cases} \frac{\sum_{v \in V} \hat{\mathsf{CC}}\mathsf{max}(V) - \hat{\mathsf{CC}}}{|V|-1} & : \min_{v \in V} \mathsf{CC}(v) < 1 \\ 0 & : \min_{v \in V} \mathsf{CC}(v) = 1 \end{cases} \in [0,1] \tag{7.10}$$

  with $\hat{\mathsf{CC}}\mathsf{max}(V) = \mathsf{max}_{v \in V} \hat{\mathsf{CC}}(v)$, $\hat{\mathsf{CC}}(v) = 1 - \frac{1-\mathsf{CC}(v)}{1-\mathsf{min}_{v \in V}\mathsf{CC}(v)}$, and $|V| > 1$. In case of $\min_{v \in V} \mathsf{CC}(v) = 1$ all vertices have the same $\mathsf{CC}(v)$, and thus, there are no central vertices. Otherwise the deviation from the maximum $\hat{\mathsf{CC}}\mathsf{max}(V)$ of $\hat{\mathsf{CC}}(v)$ of each vertex is summed up and normalized by $|V|-1$. The larger proportion of vertices with the minimal sum of geodesic distances to all other there are, the smaller the $\mathsf{CC}(G)$ value is. Conversely, if there is only one vertex

[15] The DC is 1 for a *star graph* (i.e., all vertices have $d = 1$, and one vertex has $d = |V| - 1$).

with a short distance to all the others, and the others are maximally distant to each other, then $\mathsf{CC}(G) = 1$.

- *Graph centrality* (GC) (*Hage and Harary*, 1995) is an alternative distance based measure that defines centrality considering the maximal geodesic distance of vertices (and not all short distances compared to the maximum as the CC).[16] For each vertex $\mathsf{GC}(v)$ is calculated as follows:

$$\mathsf{GC}(v) = \frac{1}{\max_{w \in V} \mathsf{gd}(v, w)} \in [0, 1]. \tag{7.11}$$

Mehler (2008a) presents a graph related index based on $\mathsf{GC}(v)$ which is computed in analogy to CC:

$$\mathsf{GC}(\mathsf{G}) = \frac{\sum_{v \in V} \hat{\mathsf{GC}}\mathsf{max}(V) - \hat{\mathsf{GC}}}{|V| - 1} \in [0, 1] \tag{7.12}$$

with $\hat{\mathsf{GC}}\mathsf{max}(V) = \mathsf{max}_{v \in V} \hat{\mathsf{GC}}(v)$ and $\hat{\mathsf{GC}}(v) = 1 - \frac{1 - \mathsf{GC}(v)}{1 - \mathsf{min}_{v \in V} \mathsf{GC}(v)}$.

7.2.6.3 Interpretation:

At first, we expect the GSDNs to have few vertices of a high and many of low centrality. Since we deal with scale-free networks, a close relation of centrality to the degree distribution becomes obvious. The fact that centrality does not exceed the value of 30% for all GSDNs holds for all centrality indices, according to the results obtained. This fact is again in line with the Zipfian distribution of dependency relations, in which a small number (of up to 30%) of vertices is central, and the rest peripheral.

Degree centrality varies between 0 and 1, taking the value of 0 if the network has the shape of a circle graph, and 1 if the network is a star graph (*Mehler*, 2008a). When transferred to GSDNs, we can expect our networks do not belong to any of these extreme cases. In case of a circle graph (i.e., $\mathsf{DC} = 0$) all vertices would have the same degree which is not given due to the power-law distribution of degrees. A star graph (i.e., $\mathsf{DC} = 1$) is also not likely to occur while only one vertex would have the highest degree, and the rest of the vertices would all have the degree of 1. In the last case, all words would have a dependency relation with only one and the same word, which is unrealistic. That is, we can expect a small number of central vertices (hubs) with a high degree centrality. Germanic languages are presumably less degree centralized (tend towards a circle graph) since many vertices of high degree centrality should be present. In contrast we expect slavic languages to have higher degree centralities. However, typological differences between languages within the families can also be expected.

The graph centrality is 1 in a graph which has the lowest maximal geodesic distance among all vertices (*Mehler*, 2008a). For a language GSDN this means to have

[16] See *Mehler* (2008a).

words which are repeated in every sentence, which is as noted above, is very unlikely. We can imagine a simple language that consistently uses the same words to build a sentence. If we were dealing with only one text, for example, a biographical note on the author Puškin, Puškin could appear in almost every sentence.[17] As a result, this would increase the centrality of the graph. In contrast, in our case we expect low graph centralities.

7.2.7 The Distribution of Components

7.2.7.1 Description

The coefficient $lcc(G)$ relates the number of vertices W of the *largest connected component* (lcc) to the total number of vertices V of the graph. The value of lcc(G) is 1 if all vertices are connected within the same component, and ~ 0 in the case of a disconnected graph of a high order.

7.2.7.2 Definition

The lcc(G) is computed as follows:

$$lcc(G) = |W|/|V| \in [0, 1]. \tag{7.13}$$

7.2.7.3 Interpretation:

What does it mean for GSDNs that all vertices belong to the same component? Assuming that the number of connected components grows towards $|V|$. This would mean that the probability of encountering the same word in proceeding sentence by sentence is nearby zero. This, in turn, would indicate an infinite number of word forms in language (i.e., each new sentence contains new words), which contradicts the sparseness and efficiency principals of language (*Ferrer i Cancho and Sole*, 2001). Natural language has a core vocabulary of function words that occur much more often than once in sentences. These words are assumably hubs of the GSDN that keep the network connected. Since all words of a GSDN are connected to each other by means of some dependency relations, and since words (especially function words) are repeatedly used in language, we expect the majority of words to be connected within the single lcc. However, it is still possible that parts of a GSDN remain disconnected. Why so? Here are some possible explanations listed:

- thematic outliers (sentences that do not contain frequent word forms but contain some thematic terms never used later on (e.g., keywords)),
- listings, numbers and compounds that only occurring the one time,
- artificiality of data (insertions of different language comments, editorial notes, etc.).

[17] Even this is not an appropriate example, since natural language in general avoids repetitions of names using, e.g., anaphoric expressions.

In general, we can look at the number of components and ask for each GSDN why there are so many connected components present in it. Is this a typological particularity of the language or is this due to the artificial composition of the treebank? We will see later, when discussing the results, that both can be true. This fact, of course, biases the contribution of the coefficient for typological studies. In addition, we compute the cumulative size distribution of connected components to see how are the sizes of the components distributed (features $F10$ and $F11$).

7.2.8 Compactness

7.2.8.1 Description

Compactness (Cp) is a coefficient from classical hypertext theory introduced by *Botafogo et al.* (1992) that measures the interconnectedness among the vertices in a network. High compactness ($\mathsf{Cp} = 1$) means that each vertex is connected to all other vertices in the graph resulting in a *completely connected graph*. A fully disconnected graph, in turn, has a compactness of 0. The benefit of this measure, as stated by *Botafogo et al.* (1992), is its independence from the size of the network, allowing to compare networks of different or equal size structurally (however, this statement does not hold for all kinds of graphs, see below).

The central question associated with compactness is: *how interrelated are the vertices of the network?* Centrality measures can be computed only for connected components, whereas Cp integrates also disconnected parts of the graph. Thus, we expect additional information on the overall structure of GSDNs by considering Cp.

7.2.8.2 Definition

Compactness of *Botafogo et al.* (1992), reformulated by *Mehler* (2008a) can be computed for a graph as follows:

$$Cp(G) = \frac{(Max(G) - (\sum_{v \in V} \sum_{w \in V} gd(v,w) + D_{max}(G) \sum_{G' \in Com(G)} |G'||V| - |G'|^2))}{Max(G) - Min(G)} \in [0,1] \tag{7.14}$$

with $Max(G) = D_{max}(G) * (|V|^2 - |V|)$, $Min(G) = (|V|^2 - |V|)$ and $Com(G)$ as the set of connected components of G. Furthermore, $D_{max}(G)$ is the maximal value of a diameter of a linear graph of order G summed to 1 (this is done in order to set a distance for disconnected nodes that is larger by one than the maximally possible distance).

7.2.8.3 Interpretation:

Compactness is 0 for a completely disconnected and 1 for a completely connected graph. As noted in (*Mehler*, 2008a) $\mathsf{Cp}(G)$ is related to the features $\gamma(S)$ and to lcc but it is assumed to contain more information about the graph than just that on its size (cf. Sec. 7.3.4 for a detailed analysis of Cp). Even in the case of a single connected component not all vertices must be connected. Cp captures the internal structure of the graph, that is, for the same *lcc* value Cp may differ reflecting the degree of connectivity of all vertices within a graph.

Transferred to GSDNs, the Cp of 1 means that all pairs of words are equally likely to be connected. This is a strong presupposition that contradicts the dependency principal stating that hierarchical syntactic organization results in selective attachment of vertices. Of course, in a language like English, where the same word forms can function as different POS the compactness could approximate 1 when the size of the treebank is sufficiently large. However, the Cp of 1 (i.e., complete connectivity of vertices) is highly improbable due to selective connectivity in GSDNs. Though, in general we expect the Cp values to be high in hierarchical networks that are known for short distances between vertices, that is, are compact (*Alava and Dorogovtsev*, 2004).

The distribution of components for networks explored in (*Mehler*, 2008a) was similar for all types of networks resulting in similar compactness values. Wiki graphs, for example, all had a power law distribution of connected components consisting of one largest component and a few smaller ones. In order to have a closer look at the internal structure of this component, (*Mehler*, 2008a) developed the measure Cp_{lcc} and applied it to the lccs of the graphs. It turned out that the last measure was more informative than Cp in reflecting differences in the internal organization of the lccs of wiki graphs.

Cp is assumed to distinguish languages with respect to patterns emerging when words are grouped into sentences by means of dependency relations. Alternative measures reflecting compactness are Cp_{lcc} applied to the largest connected component (*Mehler*, 2008a) and the measure of cohesion (Ch) (see *Mehler* (2008a) for more details on these measures).

7.2.9 Cohesion

7.2.9.1 Description

The measure of cohesion Ch can be regarded as an alternative measure of compactness (see *Mehler* (2008a)).

7.2.9.2 Definition

It is defined for an undirected graph as the fraction of all edges in the graph to the number of edges in a completely connected graph:

$$\mathsf{Ch}(G) = \frac{\sum_{v \in V} d(v)}{|V|^2 - |V|} \in [0, 1] \tag{7.15}$$

7.2.9.3 Interpretation:

Since the GSDNs are far from being completely connected (see the Zipfian distribution of degrees) we expect rather small values of Ch for all languages. However, there can be typological and genealogical differences between languages, as already has been outlined for the average degree.

7.2.10 Stratum

7.2.10.1 Description

The measure of stratum St, developed by *Botafogo et al.* (1992), measures the deviation of some directed graph from a linear directed graph of the same order. St is intended to evaluate the linearity of hypertext structures which are defacto represented by directed graphs.

7.2.10.2 Definition

Stratum is defined as a fraction of graph prestige to graph prestige of a linear graph of the same order. The index assumes a directed distance matrix DM. We calculate $\forall v \in V\ a_v = \sum_{u \in V} d(v,u)$ which is the sum of directed finite distances $d(v,u)$ from v to all other vertices, and $b_v = \sum_{u \in V} d(u,v)$ which is the sum of finite directed distances to v from all other vertices. The *prestige* of v is defined as $p_v = a_v - b_v$ (see also *Harary* (1959)). The prestige can be positive, negative or zero. The *absolute prestige* of a graph G is defined as the sum $AP = \sum_v^{|V|} |p_v|$. Now stratum can be calculated as

$$\mathsf{St}(G) = \frac{AP}{LAP} \in [0,1] \tag{7.16}$$

where LAP is the absolute prestige of a linear graph of the same order (*Botafogo et al.*, 1992):

$$LAP = \begin{cases} \frac{n^3}{4}, & \text{if } n \text{ is even} \\ \frac{n^3-n}{4}, & \text{if } n \text{ is odd.} \end{cases}$$

7.2.10.3 Interpretation:

Since stratum is defined strictly for directed graphs and we deal with undirected GSDNs, we did not apply this measure in our study. Another possibility to compute stratum is to relate the diameter of a graph to the diameter of the linear graph of the same order. This approach is presumably less precise than the stratum of *Botafogo et al.* (1992) (not all distances of the graph are considered), however, it can also serve as an indicator of the linearity of the graph. One benefit of considering the second variant of stratum is the possibility to deal with undirected graphs. This variant can be calculated as follows:

$$\mathsf{St}_2(G) = \frac{diam(G)}{|V| - 1} \in [0,1] \tag{7.17}$$

where $diam(G)$ is the diameter of graph $G = \{V, E\}$ divided by the diameter of a linear graph of order $|V|$.

Alternatively, we could treat the GSDNs as directed and apply the stratum of *Botafogo et al.* (1992). In general, for both stratum measures we expect the values of the index to be rather small for all GSDNs. This is because GSDNs are in general small-world like graphs rather than linear graphs, and comparing large networks to linear graphs of the same order should result in small values of the coefficient.

Network	$\|V\|$	$\|E\|$	ϵ	C_{br}	C_{ws}	lcc	L	r	γ	$\bar{R}^2$	$\gamma_{\bar{k}_{nn}(k)}$	$\bar{R}^2$	$\gamma_{C(k)}$	$\bar{R}^2$	Cp	CC	GC	DC	γ_S	$\bar{R}^2$	D	Ch	C_A
CAT	38882	215308	5.5374	0.0098	0.231	0.9978	3.0579	-0.171	-1.4616	0.9988	-0.5971	0.9726	-0.5355	0.9388	0.9957	0.2629	0.0565	0.2679	-2.1158	0.9839	9	0.0002	0.3382
ITA	7984	24269	3.0397	0.0122	0.1414	0.9898	3.412	-0.2265	-1.6307	0.9954	-0.5435	0.8412	-0.5699	0.7919	0.9795	0.2026	0.037	0.1901	-1.9923	0.9956	11	0.0007	0.3039
RUM	8867	23901	2.6955	0.0053	0.0932	0.9981	3.4466	-0.1862	-1.801	0.9947	-0.5563	0.81	-0.6024	0.7949	0.9961	0.2081	0.0451	0.2297	-1.8073	0.9784	12	0.0006	0.2861
SPA	17101	56911	3.3279	0.0069	0.1788	0.9804	3.1749	-0.1815	-1.6785	0.9966	-0.6263	0.9431	-0.5659	0.9011	0.961	0.2679	0.0622	0.2787	-5.1055	0.9998	10	0.0003	0.3052
LAT	8326	19923	2.3928	0.009	0.0931	0.9754	3.6179	-0.129	-1.6006	0.9945	-0.4431	0.9249	-0.5713	0.8805	0.9512	0.2328	0.0507	0.1904	-2.6498	0.9996	12	0.0005	0.2868
BUL	32421	95698	2.9517	0.0055	0.1693	0.995	3.3111	-0.2025	-1.5476	0.9976	-0.5932	0.9479	-0.5921	0.9313	0.99	0.2335	0.0376	0.2057	-2.0809	0.9986	12	0.0001	0.2731
CZE	146504	696379	4.7533	0.0038	0.1342	0.9714	3.3809	-0.0817	-1.1678	0.9992	-0.6178	0.9818	-0.566	0.9454	0.9436	0.2593	0.0314	0.2376	-3.5597	0.9998	16	0	0.1992
RUS	58283	177942	3.053	0.0045	0.0883	0.9927	3.7141	-0.0975	-1.463	0.9981	-0.4888	0.9391	-0.4422	0.8688	0.9854	0.2157	0.0094	0.1565	-2.8336	0.9999	21	0.0001	0.1742
SLV	8342	20453	2.4518	0.0097	0.0946	0.9647	3.6044	-0.1879	-1.8486	0.9948	-0.554	0.8737	-0.6013	0.8263	0.9304	0.234	0.0538	0.1641	-3.3107	0.9999	9	0.0005	0.3727
DAN	19133	50858	2.6581	0.0127	0.1867	0.9876	3.3257	-0.2677	-1.4443	0.9971	-0.4866	0.9439	-0.5817	0.9178	0.9753	0.2096	0.0327	0.1429	-1.8512	0.9939	16	0.0002	0.2027
DUT	32599	112613	3.4544	0.0061	0.1367	0.9934	3.4044	-0.1956	-1.2783	0.9979	-0.6119	0.9288	-0.5486	0.8383	0.9868	0.2066	0.0371	0.1814	-2.777	0.9973	11	0.0002	0.3054
SWE	25097	126526	5.0414	0.0309	0.2629	0.9943	3.1386	-0.2345	-1.153	0.9987	-0.408	0.9183	-0.3787	0.7891	0.9885	0.2167	0.0379	0.1496	-3.6197	0.9998	11	0.0004	0.2821
ENG	44748	316274	7.0678	0.0211	0.2121	0.848	3.0597	-0.1788	-1.4974	0.9995	-0.5195	0.9668	-0.4705	0.9112	0.7191	0.2406	0.0354	0.1719	-7.4715	0.9999	12	0.0003	0.1833
GER	72630	329868	4.5417	0.0109	0.189	0.9967	3.2552	-0.2082	-1.2575	0.999	-0.5122	0.9348	-0.5341	0.9029	0.9935	0.2166	0.0278	0.1316	-2.7336	0.9991	10	0.0001	0.3234
TUR	19386	28671	1.4789	0.0052	0.0128	0.8549	5.0249	-0.0721	-2.0175	0.9952	-0.3059	0.6487	-0.3313	0.5801	0.7307	0.1368	0.025	0.0415	-2.4636	0.9984	17	0.0001	0.216
JAP	3271	19072	5.8306	0.0589	0.3017	0.959	2.7606	-0.2588	-1.2724	0.9976	-0.4288	0.9441	-0.4398	0.8227	0.9192	0.2851	0.1253	0.3007	-0.6092	inf	6	0.0035	0.4231
GRC	11521	29607	2.5698	0.0111	0.0697	0.9859	3.7922	-0.1078	-1.6506	0.9956	-0.4221	0.9007	-0.491	0.7849	0.9718	0.2253	0.0664	0.1471	-2.2934	0.9941	10	0.0004	0.3686

Table 7.2: The feature vectors used for classification. SLV: Slovene, SPA: Spanish, SWE: Swedish, ITA: Italian, RUM: Romanian, DUT: Dutch, BUL: Bulgarian, CAT: Catalan, DAN: Danish, RUS: Russian, CZE: Czech. Black triangles in the lower row point at features from one of the eight-best-off-combinations.

7.3 Typological Interpretation of the Results obtained for GSDNs

In this section we discuss the values of the network indices computed for GSDNs.[18]

7.3.1 lcc

Following our interpretation of the coefficient lcc (see above in Sec. 7.2.7), we can expect that GSDNs in general result in a single largest component. Looking at the results of the coefficient, we can see that it is indeed true for all languages that the majority of words is connected within the largest component (lcc $\sim$ 0.9). However, in some languages the majority of components contains more than one word. In this section we will look closer at these languages, and try to find explanations for this phenomenon. As mentioned above in Section 7.2.7, at least three criteria can be outlined that may have caused the presence of more than one component in a GSDN. These criteria are: thematic diversity, numbers, listings and other structural elements as well as the artificiality of data. Inspecting the languages we will take these criteria into consideration.

According to the results in Table 7.2, there are two languages with an lcc below 0.9 - English and Turkish. The English treebank contains a large amount of numbers (142.15, 42.5, etc.) which are attributed to, for example, the amount of money. The texts stem from newswire and, thus contain many of these numbers. The numbers are given a self link but no other dependency relations within the sentences. Thus, such numbers contribute to the increase of connected components in the English GSDN. Obviously, this is rather the artificiality of data that results in a lower value of lcc.

Turkish is an agglutinated language, thus, instead of speaking of words, we rather focus on morphemes, which are also annotated with quasi-syntactic structure. Words (that resemble phrases) and morphemes in Turkish are much less common than words in other languages, thus, isolated components of small size can increase the number of components, and consequently, decrease the value of lcc. In addition, we find isolated words in the Turkish treebank to be foreign words such as "the", which are attached to the virtual root of the sentence and have no other relations to other morphemes. But there are also single words or expressions in Turkish (like *Hayri_Baytas'tan* and *alinti*) that are connected to each other, while neither being connected to the remaining parts of the sentence, nor to other vertices in the GSDN. Here too, the annotation scheme influences the formation of components. That is, some isolated morphemes or quasi-phrases are attached to the virtual root, rather than to the root of the sentence (e.g., the predicate). Since we do not consider the virtual root constructing the GSDN (because the virtual root is not a lexical element), these items remain unattached. That is, the specifics of annotation, which could be expected for Turkish, do not influence the structure of the GSDN according to lcc.

Japanese also has a low value of lcc, though, slightly higher than 0.9. In taking

[18]Note that not all measures presented in the previous section were computed for GSDNs. This reduction was made in order to spare computation costs and to keep the feature space clear.

a closer look at this language, we were able to discover many foreign elements of the German language present in the treebank. These elements are isolated and they form isolated components.

Concerning lcc, we can conclude that this feature can be biased by the specifics of annotation, formation and content of the treebank.

7.3.2 Clustering

As mentioned in the previous sections, we consider two clustering coefficients to measure the clustering of GSDNs. Figure 7.5 illustrates the different behavior of the two coefficients. Although both coefficients correlate ($corr = 0.717, p = 0.012$)[19] positively, the correlation is not linear. As shown in Figure 7.5, the values of C_{br} behave differently for many languages.

For a better understanding of the two coefficients' behavior, we refer to the work of *Soffer and Vázquez* (2005) who point to the influence of degree correlations (assortative vs. disassortative) on the values of C_{ws} and C_{br}. The local clustering coefficient c_i (see Equation 7.2) of the vertex i weights the number of edges among its neighbors by the number of maximally possible connections among them according to the degree of i. *Soffer and Vázquez* (2005) criticize this approach since only the degree of i is considered, and not the maximally possible degrees of its neighbors. The neighbors may have smaller degrees than k_i reducing the number of possible triangle relations. That is, if the neighbors have degrees $\geq k_i$, the normalization by $k_i(k_i - 1)/2$ is appropriate, if however the degrees of the neighbors are smaller, then a neighbor-degree-sensitive normalization is to be applied in order to rule out the dependence on degree correlations (*Soffer and Vázquez*, 2005). *Soffer and Vázquez* (2005) propose a new measure of local clustering taking the above issues into account. They estimate the number ω_i of the maximally possible connections between the neighbors of i, and define the local clustering as follows:

$$\tilde{c}_i = \frac{e_i}{\omega_i} \tag{7.18}$$

Using this vertex related measure they compute $\tilde{C_{ws}}$ and $\tilde{C_{br}}$ and compare the results to original C_{ws} and C_{br} values for four real graphs with different degree correlations (from highly disassortative to highly assortative). The results (listed in Table 7.3) show that C_{ws} and C_{br} highly diverge for disassortative graphs, while less so for assortative graphs, which is not the case when the degree correlations are eliminated ($\tilde{C_{ws}}$ and $\tilde{C_{br}}$).

Indeed, the values of $\tilde{C_{ws}}$ and $\tilde{C_{br}}$ (cf. Table 7.3) are nearly the same for the two disassortative graphs (Internet (Int) and Protein (Pro) networks)[20] ($\tilde{C_{ws}}$(Int) = 0.49

[19] We have computed the coefficient of correlation *corr* using MATLAB version 7.2.0.232 (R2006a) and Curve Fitting Toolbox (www.mathworks.de) in order to obtain the pairwise correlations among all features obtained for 17 GSDNs. The correlations are significant if $p < 0.05$.

[20] The two graphs referred to here are (1) the autonomous system representation of the internet for April 2001 (*National Science Foundation*, 2001) and (2) the protein-protein interaction network of the yeast *Saccharomyces cerevisiae* (*UCLA and Eisenberg*, 2010).

Network	r	$\langle c \rangle$	C	$\langle \widetilde{c} \rangle$	$\widetilde{C}$
Internet	−0.19	0.45	0.0090	0.49	0.45
Protein interaction	−0.13	0.12	0.055	0.16	0.19
Semantic	0.085	0.75	0.31	0.83	0.59
Co-authorship	0.67	0.65	0.56	0.78	0.85

Table 7.3: Average clustering coefficients for graphs with varying degree correlations listed in increasing order of assortativity (*Soffer and Vázquez*, 2005, 1).

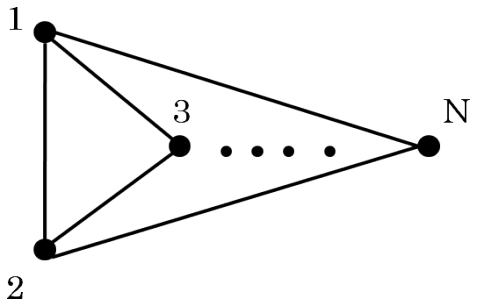

Figure 7.4: Example: a double-star graph (*dsg*) (*Soffer and Vázquez*, 2005, 1).

vs. $\tilde{C_{br}}(\text{Int}) = 0.45$ and $\tilde{C_{ws}}(\text{Pro}) = 0.16$ vs. $\tilde{C_{br}}(\text{Pro}) = 0.19$), what the authors did not mention is that C_{ws} also takes nearly the same values for these two graphs ($C_{ws}(\text{Int}) = 0.45$ and $C_{ws}(\text{Pro}) = 0.12$), although they highly differ from the values of C_{br} ($C_{br}(\text{Int}) = 0.0090$ and $C_{br}(\text{Pro}) = 0.055$).

The comparable values of C_{ws}, $\tilde{C_{ws}}$ and $\tilde{C_{br}}$ could have occured coincidentally. However, if we think about the structure of disassortative graphs, in which the majority of vertices of low degree connect to few vertices of higher degrees, then the maximal number of neighbor-connectivity is $\sim k_i(k_i - 1)/2$. This means, if we take the average as the global graph related aggregation function (C_{ws}), the c_i's of the low degree vertices prevail resulting in the same values of the global clustering coefficient for all variants, with and without degree correlation biases.

The picture changes for C_{br}. Here the bias of degree correlations increases even more since the sum $\sum_i k_i(k_i - 1)/2$ is considered as the normalization factor. In the case of high-degree vertices, the expected number of triangles is high, but since these vertices have mostly low-degree neighbors, the actual number of triangles is small even if all low-degree neighbors are connected to each other. A good illustration of this problem is given by the double-star graph in Figure 7.4.

Two connected high-degree vertices 1 and 2 connect to $N - 2$ low-degree vertices that all reach their maximal number of triangle relations (with 1 and 2). In case of vertices 1 and 2, the local clustering coefficient c_i (with degree bias) approaches zero for $c_1 = c_2$ if $N \gg 1$. This circumstance is not a problem for C_{ws}, since both

zero-values are ruled out by the $N-2$ non-zero c_i's (i.e., $c_i = 1$ for $i \in \{3, ..., N\}$):

$$c_1 = c_2 = \frac{e_i}{k_i(k_i-1)/2} = \frac{(N-2)*2}{(N-1)(N-2)} = \frac{2}{(N-1)}$$

$$c_i = \frac{1}{1} = 1 \; \forall i \in \{3, ..., N\}$$

$$\begin{aligned} C_{ws} &= \frac{\sum_{c_i}^{N}}{N} \\ &= \frac{\frac{4}{(n-1)} + (N-2)}{N} \\ &= \frac{4}{(N-1)N} + \frac{(N-2)}{N} \end{aligned}$$

$$\begin{aligned} \lim_{N\to\infty} C_{ws} &= \lim_{N\to\infty} \frac{4}{(N-1)N} + \lim_{N\to\infty} \frac{N-2}{N} \\ &= 0 + \lim_{N\to\infty} \frac{\cancel{N}(1-\frac{2}{N})}{\cancel{N}} \\ &= 1 \end{aligned}$$

That is, for graphs like the one in Figure 7.4 C_{ws} approximates 1 for large N. For C_{br}, however, the denominator is boosted with the high expected value of triangles for the high degree vertices (1 and 2). The coefficient approximates zero for $N \to \infty$:

$$\begin{aligned} C_{br} &= \frac{\sum_i e_i}{\sum_i k_i(k_i-1)/2} \\ &= \frac{2*(N-2)+(N-2)}{\frac{(N-1)(N-1-1)*\cancel{2}}{\cancel{2}} + (N-2)} \\ &= \frac{2*(N-2)+(N-2)}{(N-1)(N-2)+(N-2)} \\ &= \frac{3*\cancel{(N-2)}}{\cancel{(N-2)}N} \\ &= \frac{3}{N} \end{aligned}$$

$$\lim_{N\to\infty} \frac{3}{N} = 0$$

To summarize the above observations, we can state that C_{br} and C_{ws} are biased by degree correlations of the network. In case of networks with assortative mixing, this bias is less strong while vertices tend to have neighbors of the same or higher degree. For networks with disassortative mixing C_{br} is biased to a greater extent by the degree correlation, than C_{ws}. This means for GSDNs that we can expect more

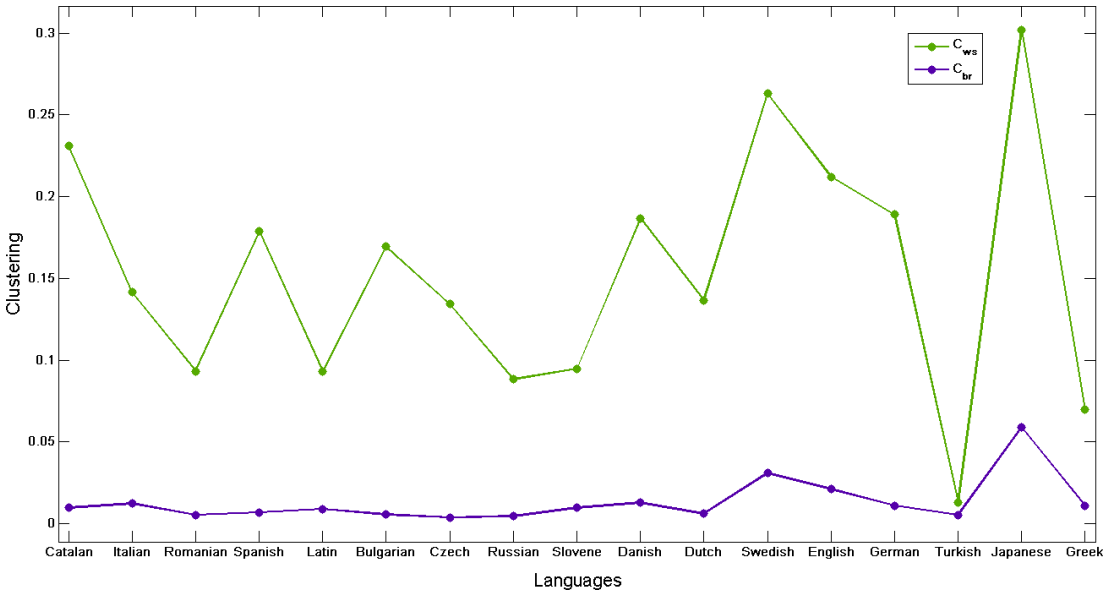

Figure 7.5: The distributions of C_{ws} and C_{br} for the 17 languages. The languages are sorted in increasing order of C_{ws}.

reliable results when evaluating clustering using C_{ws} than C_{br}. However, C_{br} still reflects the major tendencies (see Fig. 7.5). Thus, C_{br} can be informative when we compare its single values to each other.

Looking at the values of C_{ws} and C_{br} for GSDNs in Figure 7.5, we see that the values of C_{ws} are much higher than for C_{br}, as predicted by the discussion above. The differences between single languages are much greater for C_{ws} than for C_{br}, presumably C_{ws} is more distinctive here. However, the largest extremes are the same (i.e., Japanese) for both coefficients.

An interesting result is the Turkish cluster value. For Turkish, both coefficients converge having similar values. Turkish has the smallest C_{ws}-value and also one of the smallest values for C_{br}. Here, we deal with a special kind of treebank with morphemes and words as vertices. Due to a large number of different inflectional morphemes in Turkish, the treebank has a much higher number of vertices compared to other languages (see Chapter 6.5). This fact results in an almost equal number of vertices and edges. This means the network is sparsely connected and there are presumably not many triangles available. The network is constructed from morphemes and words that are very infrequent[21] decreasing the probability of triangles. These specifics are confirmed by a small largest component ($\mathsf{lcc} = 0.8549$) and by a completely different γ-value of the degree distribution ($\gamma_{\mathrm{TUR}} = -2.0175$), than for other languages ($\gamma \sim -1.5$) (see Table 7.2). Moreover, if we look at the value of the correlation coefficient r for Turkish $r_{\mathrm{TUR}} = -0.00721$, we see that the value, though negative, is very close to zero indicating that there are almost no neighbor degree correlations in the Turkish

[21] Words are infrequent since they are rather combinations of many different morphemes (words are comparable to phrases), and morphemes since there are many of them.

GSDN. This zero-correlation may explain the fact that both coefficients, C_{ws} and C_{br}, achieve nearly the same values for this treebank. Czech takes also a value of r nearby-zero, but its C_{ws} value is much higher here, so there is apparently clustering in this network. Further we have much more edges than vertices, a greater largest component, and γ-values very similar to other GSDNs. So, in the case of Czech a nearby-zero degree correlation can be presumably explained by other factors, like for instance the size of the treebank. This example vividly shows that it is important to consider many characteristics of the network in order to obtain the full picture.

7.3.3 Average Geodesic Distance

As mentioned in the previous sections, high clustering and small average geodesic distance L indicate that the respective graph fits the small-world model. Looking at the results in Table 7.2, we see that almost all the languages exhibit an average geodesic distance of $L \sim 3$. We can observe slight differences in the values for morphologically richer languages (like Russian and Latin), who have longer average distances, and analytic languages (like, e.g., English and Japanese), who have shorter distances. Japanese is an extreme example with the smallest $L = 2.7606$ and the highest clustering. Turkish is another extreme that has the longest distances on average $L = 5.0249$ and a clustering value nearby zero. These observations show that Turkish is less likely to be small-world, than the other graphs. The Turkish GSDN seems to have a distinct structure, which is not surprising if we look at the specifics of its treebank (see Sec. 6.5.16).

Turkish is an agglutinating language, and "[...] Turkish words may be formed through very productive derivations, increasing substantially the number of possible word forms that can be generated from a root word."[22] As outlined in the previous sections, vertices of the Turkish GSDN are not only words but also morphemes or so called *inflectional groups* (IGs), which increase the overall number of vertices in the network. The number of edges, however, is comparable to other GSDNs, since the grammatical dependency relations are the same. Thus, with an enormously high number of vertices, and a steady increasing number of edges (cf. the values of $|V|$ and $|E|$ for Turkish), the graph is less densely connected, which explains the high value of L. The clustering is also low, since due to a large number of vertices, the probability of two connected vertices to share a neighbor is much lower while the number of edges does not increase that much. We can conclude that the different morphological type of Turkish (agglutinating) and the different kind of annotation are responsible for the high value of L.

It would be interesting to see what the GSDN looked like, if we did not consider inflectional groups but only word forms, as normally done in dependency syntax. Here we can expect an even less structured graph with a very low number of vertices (due to the small average sentence length), and a very low lcc, since word forms in general will have a low probability to be repeated (due to a high variation within the word). When increasing the size of the treebank, however, the network could become

[22] *Eryiğit et al.* (2008, 359).

more and more connected since short grammatical words that serve as connectors also occur in Turkish.

7.3.4 Compactness

In this section we compare the results of the measure of compactness Cp to two related measures Ch and C_A. As mentioned in the previous sections, Cp correlates positively (perfect correlation) with lcc ($corr = 0.9998, p = 0$). To explain this fact, we should look more closely at the definition of Cp in Equation 7.14. Compactness is a measure that considers all the components of the graph. Cp relates the sum of all shortest paths between all vertices to the sum of maximally possible shortest paths for the graph of such order. If there is no connection between some of the vertices (i.e., they are in two different components), then, the length of the path is set to $N - 1$ (i.e., the maximal diameter for a linear graph of the order N). Therefore, the higher the number of components in the graph is, the higher the number of shortest paths of the length $N - 1$ will be, the less compact the graph will be. This explains the high positive correlation between Cp and lcc. English and Turkish, for example, have the highest number of components (i.e., the lowest lcc) and the lowest values of Cp. Unsurprisingly, compactness also correlates positively with the γ_S value of the distribution of connected components.

When we compare the values of Cp to the corresponding related measures of compactness Ch and C_A it reveals no correlations between the last two measures and Cp. This is certainly a good result, the measures seem to behave differently bearing new information about the graphs. C_A or the *relative graph connectivity* (*Mehler et al.*, In preparation) was proposed as an alternative to Cp in order to achieve a more reliable normalization of the actual number of shortest paths to the sum of maximally possible paths defined in terms of diameter. C_A is defined as follows:

$$C_A(G) = \frac{\sum_{v \in V} \sum_{w \in V} dg(v, w)}{diam * N(N-1)} = \frac{L}{diam} \tag{7.19}$$

where L is the average geodesic distance and $diam$ the diameter of graph $G = \{V, E\}$ with $N = |V|$. In contrast to Cp, the non-existing distances are not considered in the above equation, and the normalizing denominator is smaller, considering the diameter of the actual graph, and not the diameter of a linear graph of order N. That is, the measure is not sensitive to the number of components in the graph. C_A correlates negatively (perfect correlation) with the diameter D ($corr = -0.8765, p = 0$). This means, the smaller the diameter in the graph is, the higher the compactness comes out. However, there are no correlations with L, although, the measure is part of the equation. Looking at the single values of C_A we see that Japanese has the highest value just in line with the small-world characteristics (L and clustering). The smallest value of C_A is given to Russian, and not to English as in the case of Cp. That is, the influence of the number of connected components is less valuable in the case of C_A.

The measure of cohesion Ch correlates positively with C_A ($corr = 0.584, p = 0.0138$), both coefficients correlate negatively with the diameter, the coefficient of

determination of the distribution of connected components $R^2_{\gamma_S}$, and positively with GC. In addition, Ch perfectly correlates (positive correlation) with C_{br}).

In sum, Cp behaves differently to Ch and C_A elucidating various aspects of the graphs. Ch and C_A are correlated, though Ch has more correlations to other features making it less informative for characterizing the graphs.

7.3.5 Distributions

In this section we look more closely at the power-law distributions of degrees (Features F_8 and F_9), neighbor-degrees (Features F_{12} and F_{13}), clustering coefficients (Features F_{14} and F_{15}) and connected components (Features F_{10} and F_{11}).

The degree distributions look very similar for all languages, the fits are almost perfect, though the slope of the distribution (i.e., γ) can slightly deviate. The difference is valuable especially for Turkish ($\gamma_{\text{TUR}} = -2.0175$), which is due to the distinct structure of the Turkish treebank as discussed in the previous sections.

The difference is even stronger for the distributions of nearest neighbors and clustering coefficients. Turkish is the only language that results in bad power-law fits (cp. the values of $\bar{R}^2_{knn}$ and $\bar{R}^2_{C(k)}$).

The size distribution of connected components is very heterogenous for all languages, which can be attributed rather to the specifics of a particular treebank, than to typological differences. The feature γ_S correlates positively with *lcc* and Cp, as it could be expected from the discussions of the previous sections.

7.4 Summary

In this chapter, we discussed a range of indices that can be used to characterize GSDNs (and networks in general). We presented the results of computing the indices for our GSDNs and analyzed the findings.

It turned out that some coefficients are influenced by the specifics of annotation, especially by the artificiality of data. One such candidate is the index lcc. Isolation of number-terms in English, of foreign terms in Japanese, or attachment of foreign words in Turkish to the empty virtual root (i.e., the virtual root is not considered in GSDNs) – all this artificially increases the number of components in a GSDN. Thus, a small lcc does not necessarily reflect a typological or genealogical property of the language, but rather a certain bias of the data. On the other hand, this index can be used to trace back peculiarities of annotation when we aim, for example, to unify some treebanks on the level of annotation theory (i.e., Level 2, cf. Chapter VI).

We could figure out a language that stands out with respect to its values of the network features: Turkish. Due to its distinct morphological type (agglutinating), this language required a different kind of syntactic annotation to other languages. The dependency structure was assigned to words and morphemes resulting in a high number of rare words. This circumstance produced a structure of the GSDN very different from other languages which was identified by the indices of clustering, average geodesic distance, lcc, etc. Typologically, this finding has little relevance; from

the point of view of treebank creation, however, such indices can help to optimize the annotation scheme for languages of different morphological types and to correct the annotation errors.

In general, although some network indices (like, e.g., indices of centrality) express nearly the same characteristic of the graph, the analyses have shown that the features reflect slightly different aspects of the graph. Used in combination with others, these features form a reliable picture of the graph structure and of the language represented by this graph. There are also some unexpected correlations between the indices that could be figured out performing an analysis of correlation. Features that are correlated with many other features are presumably less informative in classification. Those pairs of features that are correlated with the same other features as well as with each other are redundant. One of them can be discarded for the final classification.

CHAPTER VIII

Genealogical Classification Experiments

8.1 Introduction

In this chapter we present the results of applying network based indices to genealogically classify languages into language families. In addition we test the performance of two alternative approaches to language classification, one based on n-grams (Sec. 8.2.1) and the other based on quantitative typological indices (Sec. 8.2.2) from *Altmann and Lehfeldt* (1973). The results of the experiments show that the QNA approach clearly outperforms the two alternatives as well as the baselines (of randomly assigning languages to clusters). The main contributions of this chapter are summarized in Section 8.7.

8.2 Two Alternative Approaches to Automatic Language Classifications

In this section we discuss two approaches to automatically determine and classify languages as alternatives to QNA. We evaluate their performance with respect to genealogical classes.

8.2.1 Language recognition: the NG-approach

The NG classification scenario applied here is taken from *Cavnar and Trenkle* (1994). The main idea behind this approach in *Language Recognition* studies (LR)[1] is that languages use some character sequences or N-grams more frequently than others (*Cavnar and Trenkle*, 1994). Thus, these sequences can be used as indicators to classify languages. The overall classification procedure implemented here is the following:

1. collect all n-grams present in a treebank

2. rank them according to their frequency of occurrence

[1] Cf. Section 8.2.1.

Feature	Short Description	Ling. area	Aggregation functions
Si	synthesis index (*Skalička*, 1935)	morphology	-
Dm	dependency measure (*Altmann and Lehfeldt*, 1973)	dep.syntax	μ, σ, H
Cn	centrality (*Andreev*, 1967)	dep.syntax	μ, σ, H
Sd	sentence depth (*Altmann and Lehfeldt*, 1973)	dep.syntax	μ, σ, H
Sw	sentence width (*Altmann and Lehfeldt*, 1973)	dep.syntax	μ, σ, H

Table 8.1: Typological features from *Altmann and Lehfeldt* (1973). The last column lists the aggregation functions applied to the corresponding feature: μ - arithmetic mean, σ - standard deviation, H - entropy

3. build a vector of the length k containing the most frequent n-grams[2] for each treebank

4. classify the treebanks by means of a classification algorithm (e.g. the one described in Sec. 8.3.1).

8.2.2 Quantitative Typology: the QT-approach

Quantitative Typology starts from the view on language as a system with interrelated components, that is, from a *holistic* view (see Sec. 2.2). Consequently, approaches in this field aim to find correlations among different components or typological characteristics in order to obtain insight into language structure (*Greenberg*, 1966).

Altmann and Lehfeldt (1973) propose a range of quantitative indices that allow processing of different linguistic levels (morphology, syntax, etc.) quantitatively. *Altmann and Lehfeldt* (1973) argue that the implementation of all the features should provide an adequate picture of a language. However, the lack of fully annotated language resources (e.g. with inflectional segmentation) needed to calculate the indices, complicates this task. We calculate some of the features proposed by *Altmann and Lehfeldt* (1973) (see Tab. 8.1 for an overview of these features) that are applicable to dependency treebank data. We try to classify languages by means of these features. The goal is the same as in the case of NG: to check whether this approach (henceforth abbreviated by 'QT' – quantitative typology) in comparison to QNA.

The indices from (*Altmann and Lehfeldt*, 1973) are shown in Table 8.1. These selected indices yield expressive potential in the areas of morphology and (dependency) syntax.[3]

The *synthesis index* is a quantitative feature applied to the whole treebank. The other features are tree-related, that is, they are calculated for each observation of a dependency tree. Aggregation functions are applied to these observations in order to get a single value of each index characterizing a treebank. Here, we use three

[2]We selected 300 n-grams as suggested by *Cavnar and Trenkle* (1994) for $n = \{1, ..., 6\}$.

[3]We selected these indices since they could be applied to our sort of data, i.e. dependency treebanks.

Language	Mean	STD
RUS	1	.022
RUM	.112	.015
BUL	.1	.038
DAN	.073	.012
DUT	.073	.037
CZE	.068	.009
SLV	.066	.012
SWE	.061	.007
ITA	.044	.006
CAT	.04	.007
SPA	.038	.011

Table 8.2: Mean and standard deviation (STD) values of Si for each treebank averaged over 29 text samples, each of which contains 1000 words.

aggregation functions to aggregate single observations of each value of an index in a treebank: arithmetic mean (μ), standard deviation (σ) and the entropy (H) (Fig. 8.1, last column).

8.2.2.1 Synthesis Index

The synthesis index of *Skalička* (1935) is attributed to the morphological complexity of a language. For a sample (dependency treebank) the synthesis index Si is calculated as the number of sentences $|S|$ divided by the number of words $|W|$:

$$Si = \frac{|S|}{|W|} \tag{8.1}$$

Skalička (1935)'s is the simplest index since it does not require a morphological analysis of treebanks (*Altmann and Altmann*, 2005). This index allows to assign a value to the language on the analytic vs. synthetic scale of morphological complexity. It ranges between $[0, 1]$ if $|S| < |W|$. The higher is the index, the more synthetic the corresponding language is. However, some authors (*Altmann and Altmann*, 2005) claim that the index is inappropriate for typological studies due to its high variability. In fact, Si is the reciprocal of the sentence length, which can be influenced not only by the language type but also by other textual factors such as author style, genre, etc. Unfortunately, factors such as genres, authors, text types, etc. are not equally balanced in every treebank. However, we have extracted samples of equal size from each treebank and consider the mean and standard deviation values. From every treebank

Language	Mean	Entropy	STD
DUT	0.792	6.109	0.134
RUM	0.708	6.556	0.178
BUL	0.6	7.082	0.14
RUS	0.538	7.985	0.158
CZE	0.512	8.721	0.177
DAN	0.488	8.378	0.161
SWE	0.47	8.896	0.146
SLV	0.461	8.673	0.132
SPA	0.402	9.622	0.163
CAT	0.382	<u>9.732</u>	0.149
ITA	0.37	9.726	<u>0.202</u>

Table 8.3: *Dm* mean, entropy and standard deviation (STD) values. The languages are arranged in decreasing order of mean. Maximal entropy and STD values are underlined.

we select 29 text samples of 1000 words each.[4]

Table 8.2 lists the *Si* values for the 11 languages. The variability of the index within the same language is rather small (see STD) for all treebanks, contradicting the prediction by *Altmann and Altmann* (2005). However, it may be the case that our randomly selected samples cover only a small range of the language internal variation, so that in fact the variability of the index is higher. As expected, Russian, being a highly synthetic language exhibits the highest $Si = 1$. Other languages feature rather small values. Typologically the results obtained by the index are realistic, although, the differences between the individual languages cannot be considered as valuable if they differ on the second or third decimal place.

8.2.2.2 Dependency Measure

This index considers the question of how many dependent elements are subordinated to the root of a dependency tree. This concerns a) the number of elements directly or indirectly depending from the root as well as b) indirect dependents on deeper levels. The *dependency measure*[5]

$$Dm = \frac{\sum_{j=1}^{m} j * x_j}{Dm_{\text{Max}}} \in [0, 1] \tag{8.2}$$

[4] 29 is the maximal number of samples with 1000 tokens that can be taken from the smallest treebank (i.e. from Slovene). That is, we select 29 as the least common number of samples for each treebank.

[5] This and the following indices were calculated on a sample of 1499 sentences from each treebank. This number is the smallest common number of sentences obtainable from each treebank.

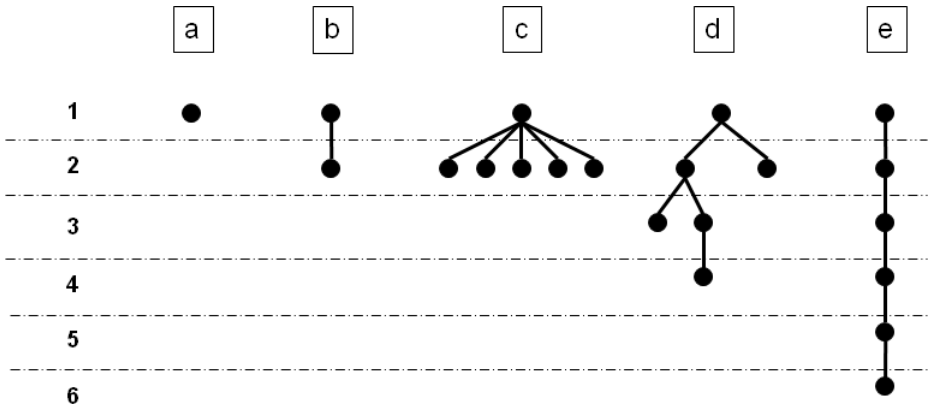

Figure 8.1: Examples of different dependency trees.

calculates this information for a single dependency tree. j is an index of levels starting from the root, that is, the root has level 1, direct children have level 2, etc. m is the maximal level of a particular dependency tree (e.g. m of graph `d)` in Fig. 8.1 is 4). j is used as a weighting factor multiplied by the number x_j of vertices on the corresponding level. Thus, the deeper the tree is, the more the vertices are weighted on deeper levels, the higher the value of Dm is. $Dm_{\text{Max}} = \sum_{j=1}^{m} j = \frac{n(n+1)}{2}$ is the Dm of a linear graph (e.g., graph (e) in Fig. 8.1). Thereby $m = n$ in this particular case, since each level of this graph has only 1 vertex.

Obviously, languages that preferably use short and flat dependency structures will have smaller Dm values than languages having complex dependency hierarchies. Typologically, this index can be highly informative if two similar languages, for example, prefer particular types of trees. However, the complexity of a dep. tree can also be an artefact resulting from the particular dependency theory rather than reflecting a certain property of a language.

Table 8.3 lists the results for 11 treebanks in decreasing order of their mean values. Obviously, there are languages that are similar to graphs like `(d)`, and others which are similar to graphs like `(c)` (in Figure 8.1). Dutch and Romanian are examples of more complex tree graphs that are closer to a linear graph than the other. Czech and Russian have presumably rather flat dependency structures (according to Dm), and Bulgarian lies somewhere in the middle. The Romance languages Catalan, Spanish and Italian occupy the lower boundary of the Dm spectrum, featuring short and simple dependencies on average. Romance languages exhibit also the highest entropy and a supposably higher redundancy of similar structures. The standard deviation is only high for Italian, presumably, there is a larger variation in the dependency structures compared to other Romance languages.

All in all, Dm allows for interesting insights into the organization of dependency relations in language. However, in order to make any serious claims, various factors such as the uniformity of dependency annotations, genre and style variations should be examined.

Language	Mean	Entropy	STD
CZE	0.344	5.138	0.325
DUT	0.29	3.197	0.311
SLV	0.221	4.66	0.289
DAN	0.206	3.489	0.31
RUS	0.204	3.005	0.367
BUL	0.159	2.592	0.265
CAT	0.129	4.554	0.2
SWE	0.126	2.835	0.25
SPA	0.116	4.329	0.196
RUM	0.116	1.409	0.309
ITA	0.108	3.933	0.19

Table 8.4: Cn mean, entropy and standard deviation (STD) values. The languages are arranged in decreasing order of mean. Maximal entropy and STD values are underlined.

8.2.2.3 Centrality

The coefficient of Cn originates from *Andreev* (1967). Here we use its modified version by *Altmann and Lehfeldt* (1973). Cn expresses the centrality of a predicate according to the linear order of the sentence. The sentence is represented as a sequence $a_l...a_3a_2a_1Pa_1a_2...a_k$ with the predicate P as its central element. k is the index running from the left-most word after P and l is the index running backwards from the right-most word before P. Thus, k and l represent the number of words on the right or left side of the predicate. Cn is computed as follows:

$$Cn = 1 - \frac{|k - l| - \delta}{k + l} \in [0, 1] \tag{8.3}$$

where δ is used to avoid impreciseness in the case of an uneven number of words in a sentence:

$$\delta = \begin{cases} 0 \text{ if k+l is even} \\ 1 \text{ if k+1 is uneven.} \end{cases}$$

A sentence $a_2a_1Pa_1a_2a_3$, for example,[6] is perfectly central, but if we omit δ we get a Cn-value less than 1:

$$Cn' = 1 - \frac{|3 - 2|}{3 + 2} = 1 - \frac{1}{5} = \frac{4}{5}.$$

Yielding δ we get the maximal centrality of 1:

$$Cn = 1 - \frac{|3 - 2| - 1}{3 + 2} = 1 - \frac{0}{5} = 1.$$

[6]The example is taken from *Altmann and Lehfeldt* (1973).

Language	Mean	Entropy	STD
DUT	0.710	4.551	0.152
RUM	0.643	5.034	<u>0.186</u>
RUS	0.505	5.848	0.143
BUL	0.504	5.061	0.128
DAN	0.465	6.031	0.157
CZE	0.458	6.452	0.176
SWE	0.411	6.507	0.139
SLV	0.403	6.337	0.123
ITA	0.386	<u>7.485</u>	0.162
SPA	0.352	7.276	0.141
CAT	0.341	7.305	0.134

Table 8.5: *Sd* mean, entropy and standard deviation (STD) values. The languages are arranged in decreasing order of mean. Maximal entropy and STD values are underlined.

Typologically, there are languages with a centralized vs. polarized syntax (e.g. Latin, Hindi, Japanese, etc.) (*Altmann and Lehfeldt*, 1973). If a language features mostly sentences of Cn nearby 1, then this language has a centralized syntax. If the predicate is likely to occur on the left or right hand side of the sentence, a language exhibits polarized syntax. Unfortunately, the index does not distinguish between right- and left-polarized types.

In our case, neither of the languages is centralized. Small differences occur between them, thereby, Spanish, Romanian and Italian have mostly polarized syntax, and Czech, for example, is rather centralized. Remarkably, Czech and Russian exhibit the highest standard deviations, which reflect presumably the languages' free word order and the variation in the predicate position. In addition high entropy for Czech and low entropy values for Romanian undermine the above observation.

8.2.2.4 Sentence Depth

Sd is the proportion of the number of levels m[7] to the number of words in a dependency sentence.

$$Sd = \frac{j_{Max}}{n} = \frac{m}{n} \in [0, 1] \tag{8.4}$$

Sentence depth arranges languages in a way similar to Dm. Obviously, these two measures are related since both take the depth of the dependency tree into account. *Sd* has nearly the same standard deviation values for all treebanks, which shows that

[7] m is the maximal value of the index j (see Dm), i.e. the deepest level in the dependency tree.

Language	Mean	Entropy	STD
ITA	0.461	<u>7.487</u>	0.173
BUL	0.458	5.019	0.142
RUM	0.434	4.741	0.172
DAN	0.415	6.076	<u>0.224</u>
DUT	0.385	4.039	0.128
SLV	0.371	6.547	0.160
RUS	0.359	5.449	0.180
SWE	0.351	6.340	0.128
CZE	0.349	5.718	0.154
SPA	0.314	7.213	0.141
CAT	0.305	7.196	0.120

Table 8.6: Sw mean, entropy and standard deviation (STD) values. The languages are arranged in decreasing order of mean. Maximal entropy and STD values are underlined.

Index	**a)**	**b)**	**c)**	**d)**	**e)**
Dm	1	1	0,52	0,71	1
Sd	1	1	0,33	0,66	1
Sw	1	1	1	0,4	0,2

Table 8.7: Values of Dm, Sd and Sw for the trees in Fig. 8.1.

the index is relatively stable. Italian, Spanisch and Catalan group together again with the lowest mean and entropy values.

8.2.2.5 Sentence Width

The Sw represents the rate of the maximal width of a single level ($W_{\max}$) to the number of elements in a dependency tree while omitting its nucleus (i.e. $n-1$).

$$Sw = \frac{W_{\max}}{n-1} \in [0,1] \tag{8.5}$$

This measure is relative to the number of elements in a sentence since a small number of elements results in a higher Sw. Sw also correlates negatively with Sd since deep sentences have mostly sparse levels (i.e. a small number of words on each level). This can be illustrated with graphs `(c)` and `(d)` in Table 8.7, in which `(c)` results in low Sd and high Sw values, and `(d)` in high Sd and low Sw values.

In comparing Sd and Sw of the 11 languages, we see that the negative correlation holds, for example, for Dutch or Russian (high Sd and low Sw). However, Spanish and Catalan have both low Sd and Sw values on average. This indicates that the two

languages have both flat and small sentences. This could be, of course, a typological peculiarity of the two languages or simply the influence of text genre (mostly newswire texts).

We calculate the above measures for all treebanks and test their combined performance in a language classification task.

8.3 Experimentation

8.3.1 Classification Scenario

To evaluate the performance of QNA, NG and QT regarding the genealogical gold standards, we classify languages by means of feature vectors consisting of feature values from one of the three approaches. The classification procedure instantiates the QNA algorithm of *Mehler* (2008a) that can be summarized as follows:

1. Initially, each input network is represented by a vector of topological indices.

2. In the next step, a genetic search is performed to find salient features within the vectors that divide the networks best according to the underlying gold standard. However, this process may stop at a local maximum so that it does not necessarily find an optimal feature subset.

3. Based on the appropriately projected feature vectors, a hierarchical agglomerative clustering is performed together with a subsequent partitioning that is informed about the number of target classes (*Mehler*, 2008a).

In summary, QNA takes the set of input GSDNs together with the parameter space of linkage methods and distance measures to find out the feature subset that best divides the data according to the underlying classification.

The application of QNA can be illustrated by using the topological indices displayed in Table 7.1 as follows:

1. In the first step we extract a set of GSDNs N from the treebanks (Sec. 6.6),

2. Then we select the network features $\mathbb{F} = F_1, ..., F_n$ (Chapter VII) and compute them for every graph G_i of the set N (Tab. 7.1),

3. Thirdly we build a feature vector $v_i = (F_1(G_i), ..., F_n(G_i))$ consisting of composite feature values for every instance of $\mathbb{F}$ in graph G,

4. Finally, we cluster the networks by means of the feature vectors.

In analogy to QNA, we compute NG- and QT-related features and input them into the classification algorithm included into QNA.

8.3.2 Evaluation

We evaluate our classifications by means of *F-measure* statistics. These statistics are usually applied in machine learning to evaluate, for example, the goodness of a classification of documents to predefined categories. They are based on two measures, *precision* and *recall*, which show (language family) the amount of correctly-classified languages for each category. *Precision* relates the number of correctly classified languages to the total number of languages classified to this group. *Recall* relates the correctly classified languages to the number of languages which are known to belong to this group. Both measures are in the range of $[0, 1]$, where 1 indicates that languages are classified perfectly and 0 that the classification failed. The F-score mediates between the two measures evaluating the overall performance of the classification. This means, if precision and recall are high, the F-score of a category is also high (i.e., close to 1). The *F-measure* is a weighted harmonic mean that considers the F-scores of all the categories. Given a known partition of languages $\mathbb{L}$ (e.g. known genealogical classes), the set of language classes $c_i \in \mathbb{C}$ and the total number of languages L, the F-measure is computed as follows (*Hotho et al.*, 2005):

$$\text{F-measure}(\mathbb{L}) = \sum_{i=1}^{|\mathbb{C}|} \frac{|c_i|}{L} F_{score}(c_i) \in [0, 1] \tag{8.6}$$

With the F-measure we get a single value between 0 and 1 which characterizes the overall success of the classification.

The results are tested compared to two kinds of baseline, one with a known-partition (i.e. the algorithm "knows" how many languages should be in each group), and the other assuming an equi-partition of languages. Using the two scenarios, languages are randomly assigned to the target categories, and the probability (in terms of F-measure) of assigning languages correctly that way is computed. This random assignment does not necessarily result in an F-measure value of 0. Its value can be understood as an expected value of assigning languages completely by chance. Thus, the classification k succeeds, if random clustering is outperformed in a way that the F-measure F_k of the classification is $F_k \gg F_{rand}$.

8.4 Experiment 1: 11 Languages

To evaluate the three competing approaches NG, QT and QNA we check whether we can successfully classify languages into 3 genetic groups: Slavic, Germanic and Romance using QT, NG and QNA. We determine the number of clusters to be equal 3 and check whether the languages are classified correctly.

A problem with the n-gram based approach (NG) concerns Russian and Bulgarian, which both use the Cyrillic writing system. This means, the n-grams of both languages cannot be directly compared to the other 9 languages using the Latin alphabet. Additionally, transliteration effort is required. In order to avoid biases due to different writing systems, we tested an additional variant of the NG-approach excluding the two Slavic languages (abbreviated by NG-RB). This addition results in four

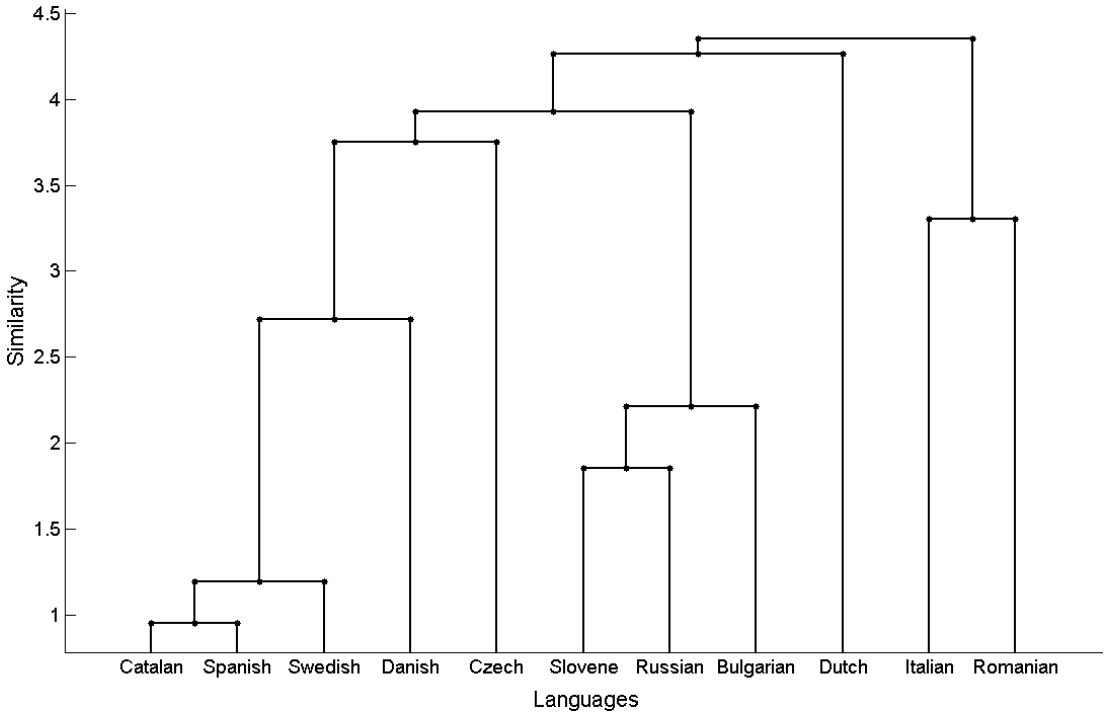

Figure 8.2: The similarity tree of languages generated by the best feature combination of QT.

different procedures that are compared to each other. The combinations performing best of each of them are visualized by means of dendrograms.

8.4.1 Results and Discussion

The results of the semi-supervised clustering experiment on 11 languages are presented in Tables 8.8-8.11.[8] The tables are structured as follows. The first column explains the clustering procedure used. The second column presents the corresponding F-measure values, and the third how many features of the total number in the setting (e.g. 8/13) were used. The best results for the three approaches and the best random baselines are listed in Table 8.13.

Figures 8.4 (i.e. QNA) , 8.2 (i.e. QT) and 8.3 (i.e. NG) display the best performing results in terms of a dendrogram. The height of a bar connecting two languages or clusters of languages shows the degree of dissimilarity. Thus, the lower the degree of agglomeration of two clusters is, the higher the similarity of the languages is (e.g. Fig. 8.4, Italian and Spanish) within these clusters.

8.4.2 QT-experiment

From Table 8.8 and Figure 8.2 we can see that only a subset of 11 of 14 features accounts for a reliable classification of languages into 3 groups (F-measure: .76389).

[8]The tables are presented in the same style as classification results in *Mehler et al.* (2010a).

procedure	F-measure	features
QT[mahalanobis,hierarchical,complete]	.76389	11/14
QT[mahalanobis,hierarchical,complete]	.76389	11/14
QT[mahalanobis,hierarchical,complete]	.76111	5/14
QT[mahalanobis,hierarchical,complete]	.65972	14/14
AVG over non-random approaches	.7372	
random baseline II	.56286	known partition
random baseline I	.54869	equi-partition

Table 8.8: F-measures of classifying 11 languages into 3 genetic groups by means of QT.

From Table 8.8 and Figure 8.2 we can see that only a subset of 11 of 14 features accounts for a reliable classification of languages into 3 groups (F-measure: .76389). That is, about 70% of languages are classified correctly. Figure 8.2 illustrates the within-cluster similarities resulting from applying the best-off-feature combination (11 features). Remarkably, the three Slavic languages Slovene, Russian and Bulgarian are classified within the same block. Romance languages, Spanisch and Catalan as well as Italian and Romanian are also pairwise similar, though not within the same Romance cluster. Swedish is grouped together with Catalan and Spanish, and Danish attaches to Swedish, though, with a greater dissimilarity (i.e., see the height of the bar). The total outliers are Czech and Dutch.

The overall similarities of languages reveal that the QT approach is able to recognize genealogical relationships of most languages used here. However, for the rest of the languages the classification fails, which can be due to several reasons; for example, the size of the treebank (large size of Czech), style and register variations, different annotations could have biased the result. Moreover, it is still possible that the observed similarities within the clusters reflect typological similarities between languages that come into the fore when considering lengths, depths, widths and centralities of sentences. A closer look at the distributions of single features (see Section 8.2.2 for a discussion) should lead to a better understanding of the typological values of the features.

We repeated the genetic search for best feature combinations ten times to see which features remain in each of the combinations. However, only one best-off combination of 11 features was able to produce the highest F-measure value.

8.4.3 NG-experiment

The NG-based classification of all the 11 languages (including Russian and Bulgarian) achieves an F-measure value of .81061.

The NG-experiment shows that we are able to classify languages perfectly when excluding both of the Slavic languages. However, adding these languages results in a drop of the F-measure. On the one hand, this is certainly a loss, since we achieve an F-measure of 1 (see Tab. 8.10) only when dealing with 9 instead of 11 languages. On the other hand, the approach would presumably perform perfectly, if Russian and

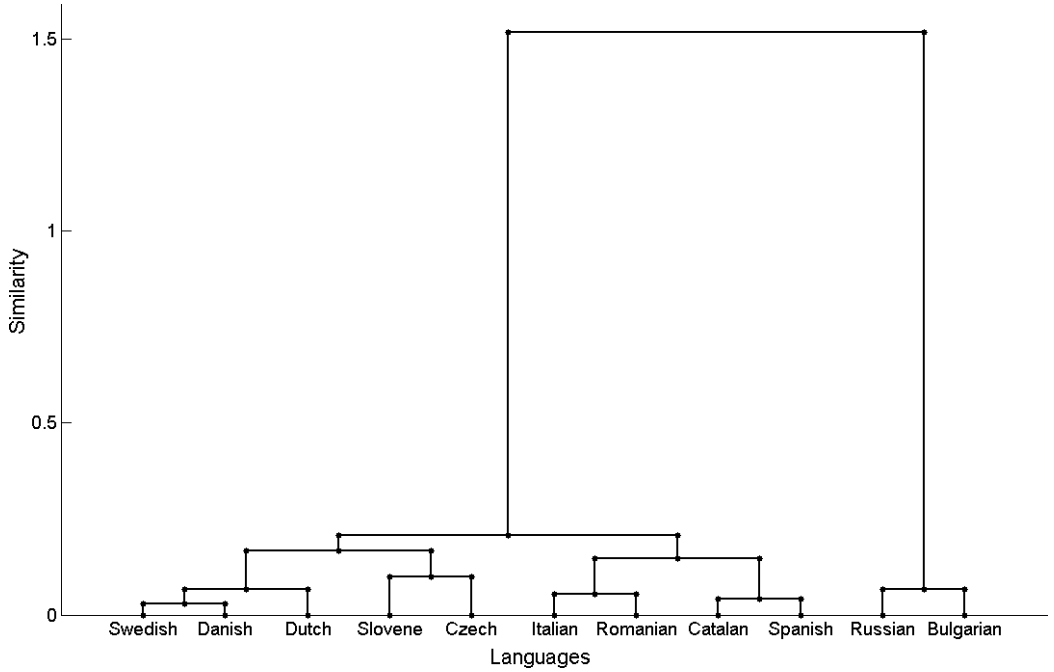

Figure 8.3: The similarity tree of languages generated by the best feature combination of the NG-experiment.

Bulgarian were transliterated into the Latin writing system. So all in all, the NG-experiment shows good performance using about 30 N-gram features. Each language family seems to have particular characters in common that are not shared (or not commonly shared) within other families.

In the context of the present study, this simple but well performing approach tells us the correct language family, however, it can also be misleading typologically, since orthographic standards in languages change, and distance between languages based solely on graphemes may be biased by orthographic peculiarities. In French, for example, the grapheme-to-phoneme correspondence is not trivial, many characters are written but not pronounced (e.g. *manquent* vs. [mãk]). This fact should influence the distance of French to other Romance languages when considering character based distances. This is just an example. However, we should be aware of such factors when we aim to go beyond identifications of language families.

8.4.4 QNA experiment

QNA also achieves the best possible F-measure of 1.0 using at least 6 features. This means, eight network characteristics suffice to separate languages perfectly with respect to genealogical relationships.

At this point, we can conclude that the genealogical classification succeeds but in order to gather more fine-grained insight into the typological information gain of the approach we have to examine single network characteristics (as done in Chap. VII). In order to see which network characteristics perform best, we have run a genetic

procedure	F-measure	features
NG[correlation,hierarchical,complete]	.81061	28/61
NG[correlation,hierarchical,average]	.81061	28/61
NG[correlation,hierarchical,single]	.80606	28/61
NG[correlation,hierarchical,average]	.80606	61/61
AVG over non-random approaches	.8083	
random baseline II	.58995	known partition
random baseline I	.5779	equi-partition

Table 8.9: F-measures of classifying 11 languages into 3 genetic groups by means of NG.

procedure	F-measure	features
NG-RB[correlation,hierarchical,complete]	1.0	36/61
NG-RB[correlation,hierarchical,average]	1.0	36/61
NG-RB[correlation,hierarchical,weighted]	1.0	36/61
NG-RB[correlation,hierarchical,complete]	.89206	61/61
AVG over non-random approaches	.9730	
random baseline II	.58426	known partition
random baseline I	.57725	equi-partition

Table 8.10: F-measures of classifying 9 languages into 3 genetic groups by means of NG-RB (n-gram based classification, Russian and Bulgarian excluded).

search for best feature combination twenty times, and received 7 best-off combinations producing an F-measure of 1.0 (see Table 8.12). The best performing features are definitely the two *adjusted coefficients of determination* of the distributions of nodes' degrees and degrees of nearest neighbours. This result is surprising since we were not expecting a good separability of GSDNs by means of these features (i.e. due to expecting a homogeneous impact of Zipf's Law). Moreover, centrality, clustering and compactness seem to be important building blocks present in each of the combinations.

The success of clustering can be explained by the loss of inflection in the particular languages. As discussed in Chapter VII, the probability of a word form to appear in many different dependency relations is higher for analytic, than for synthetic languages. This increases the probability of clusters in a GSDN. As we can see from Table 7.2, Germanic languages have clearly higher C_{ws} values than the Slavic languages. The Romance group is less homogeneous: Spanish and Catalan are close to Germanic, and Romanian and Italian to the Slavic group according to their C_{ws} values. So this feature alone is not sufficient in order to separate the languages perfectly, other features are needed to complete the picture. But C_{ws} can be used easily to examine the typological properties of languages by means of networks. Note also that in the cases where C_{ws} is not selected, C_{br} or $\gamma_{C(k)}$ with $\bar{R}^2_{C(k)}$ are applied instead.

Further, C_{ws} and L correlate strongly negative ($corr = -0.8997$)[9], which nicely

[9]We have computed the pairwise correlations among all features. Here and in the following we refer to some significant results.

procedure	F-measure	features
QNA[mahalanobis,hierarchical,complete]	1.0	7/21
QNA[mahalanobis,hierarchical,ward]	1.0	8/21
QNA[mahalanobis,hierarchical,complete]	1.0	10/21
QNA[sq.euclidean,hierarchical,weighted]	.90909	13/21
AVG over non-random approaches	.9773	
random baseline II	.56297	known partition
random baseline I	.55442	equi-partition

Table 8.11: F-measures of classifying 11 languages into 3 genetic groups by means of QNA.

points to the small world property of GSDNs that have short geodesic distances and high cluster values. Russian, for instance, has the smallest C_{ws} value and the highest value of L. Swedish and Catalan, in turn, exhibit high clustering and the shortest geodesic distances among the 11 GSDNs.

Indices of centrality tell us something about the amount of central vertices in a network. A network of few central and many *peripheral* vertices is centralized. Conversely, the DC value becomes the smaller, the more equally distributed the degrees of the vertices. In the case of GSDNs, this means to have many highly connected word forms in terms of their dependency relations. Germanic languages have lower centrality values than Slavic and some Romance languages. This means that there are much more *central* word forms in Germanic networks than in Slavic ones. It is plausible to assume that this relates to prepositions, who seem to play a greater role in Germanic languages.

The compactness remains to express the overall connectedness of a GSDN. Compactness is negatively correlated, though not significantly ($corr = -0.3015, p = 0.3676$), with L, since the larger L the less compact the graph.

In fact, many features that we used are correlated[10] (e.g. the correlation between DC and CC is positive $corr = 0.7511, p = 0.0077$, between Cp and lcc is $corr = 1, p = 0$ (perfect correlation), and between C_{ws} and L is negative $corr = -0.8997, p = 0.0002$). The positively correlated features can be easily exchanged without any loss in F-measure. If, for example, feature DC is selected, then feature CC is not needed to improve the result. The same holds for both clustering coefficients. Negative correlations mean that large values of feature X result in small values of feature Y (or vice versa), however, both features can be informative in classification (like in the case of C_{ws} and L). Cp and lcc exhibit a perfect correlation, and it becomes obvious from Table 8.12 that in 3 of 2 cases where lcc is used, Cp is used too. This is not what we would expect, one of the two features would have been sufficient for the best-off selection. However, the mahalanobis distance used for clustering ensures that the features used are statistically independent, that is, correlated features become uncorrelated in the final representation.

[10]Note that all correlations exemplified here are significant with a p-value < 0.05.

	feature			combinations					sum
	ϵ							√	1
	C_{br}			√	√		√	√	4
	C_{ws}		√			√			2
	lcc	√				√		√	3
	L		√						1
	r	√							1
	γ	√		√				√	3
▸	$\bar{R}^2_{\gamma}$	√		√	√	√	√	√	6
	$\gamma_{\bar{k}_{nn}(k)}$		√						1
▸	$\bar{R}^2_{\bar{k}_{nn}(k)}$	√	√	√	√	√	√	√	7
	$\gamma_{C(k)}$	√							1
	$\bar{R}^2_{C(k)}$	√	√						2
	Cp	√	√		√	√	√		5
	CC			√	√	√	√	√	5
	GC					√			1
	DC		√	√	√		√		4
	γ_S		√						1
	$\bar{R}^2_{\gamma_S}$	√							1
	diam					√			1
	Ch	√				√			2
	C_A	√	√						2

Table 8.12: Best feature combinations of QNA resulting in an F-measure of 1 found by the genetic search (the algorithm was run 20 times). The most frequently selected feature is C_{ws} (black triangle), followed by L, DC and Ch (transparent triangles).

8.4.5 Discussion on Experiment 1

In this section, we experimented with three different approaches to automatic language classification. We tested their performance in the task of genealogical language classification.

In the case of appliying QNA to classify languages, a network of a language was created by taking a dependency treebank of a particular language as input and mapping words of the treebank to vertices and dependency relations to edges. In this way, a language's network was constructed. Since we dealt not only with networks of different treebanks but also with networks of treebanks from different languages, the leading assumption was that particular characteristics of the language may have left their traces in the structure of the network. To account for this assumption, we calculated 21 topological characteristics on the language networks and classified languages by means of them in order to see whether the similarities in network structure reflect some "real" similarities among languages.

Indeed, we found out that some network indices allow to perfectly reproduce the genealogical relations between the languages. The network structure seems to cover several linguistic levels (i.e. morphology, syntax, lexis) and provide a more abstract, general (holistic) view on language. Furthermore, our results revealed four classes of features. Features in a class are positively correlated. They are selected by the genetic search depending on which features from other classes are selected. So, we can

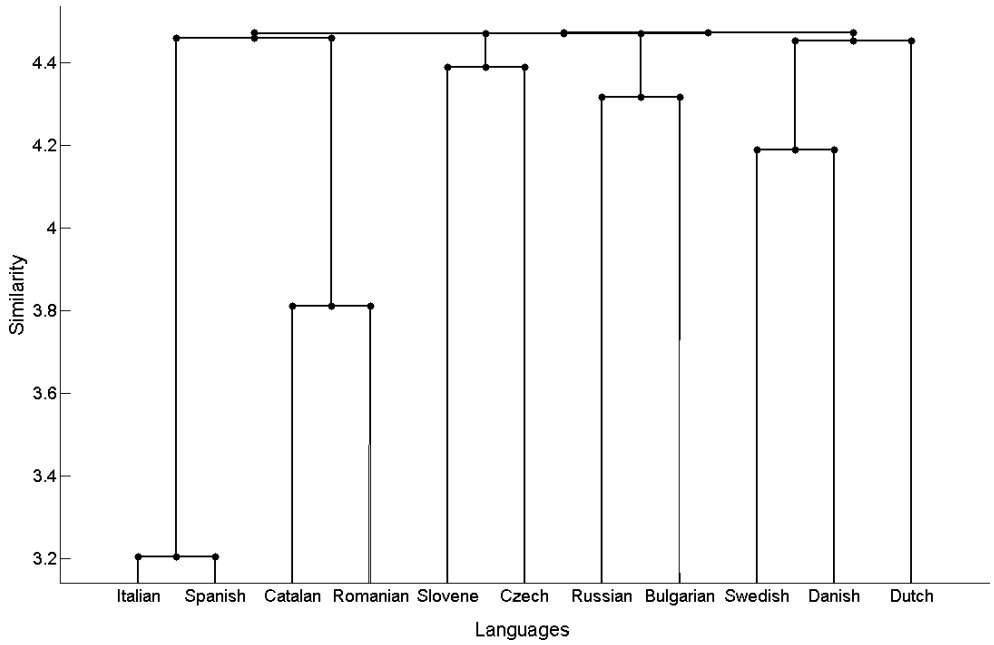

Figure 8.4: The similarity tree of languages generated by one of the best feature combinations of QNA. Best features combinations are shown in Table 8.12.

method	QT	NG	NG-RB	QNA
best non-random	.76389	.81061	1.0	1.0
average non-random	.7372	.8083	.9730	.9773
random	.56286		.58995	.56286

Table 8.13: The overall *F-measures* of the genealogical classification. The *best non-random* F-measures represent the best classification results from Tables 8.8-8.11. The *average non-random* are the average F-measures over different combinations, and (average) random F-measure values.

speak of a network of features. In future work, we aim at a systematic examination of this feature network.

The main advantage of QNA lies in the integrated view on language that enlarges the range of possibilities to examine the language as a whole system. A disadvantage of the approach is the lack of transparency with respect to the role of single features, which should be examined together with others, and also in isolation. Further, we still do not know about all the possible sources of bias (i.e., influence of genres, dependency theories, etc.). Although, the results are highly encouraging in producing a perfect classification, future work should systematically identify and eliminate possible error factors before we will be able to make judgements about the overall potential of QNA in the area of language classification.

Quantitative typological indices were computed either for single sentences or the whole treebank (sample) and used for classification. This approach proves the least

efficient in terms of F-measure. Typologically, though, QT may be interesting; why do, for instance, languages like Czech and Dutch differ from their language family members? And how can these deviations be explained with respect to features of dependency structure as explored here? However, we expect QT-features to be biased even more than the network based classification. QT-features such as sentence width, etc. can depend on the type of texts (or genre) in a sample and vary to a larger extent even within a single language. Here, we aim to examine the role of such indices within and between languages in order to facilitate the interpretation of the results.

At least the expressiveness of N-gram based classifications was confirmed again in this article. NG is typologically presumably less relevant. From the point of view of application, however, NG is more easily implemented than the other two approaches. Neither annotated treebanks nor the calculation of indices is required for NG. Transliteration may become a problem, but all in all, NG is a robust means when it comes to determining genetic relationships. When we aim to look at typological relations between linguistic levels, QNA and QT may prove a better choice.

In summary, the three approaches to automatic language classification show good performance. QNA and QT are typologically promising, though further studies should follow in order to learn more about the possibilities and limitations of these approaches.

8.5 Experiment 2: Increasing the Size of Language Families

In this section, we present the results of the genealogical classification of 13 languages (see Section 6.4 for the description of the languages) by means of QNA. In this scenario we keep the number of language families constant and enlarge the number of languages within the families. In the first variant of the experiment we add English and German to the Germanic group. This results in the combination of language groups presented in Table 8.14. In the second variant of the experiment we add Latin

	Germanic	**Slavic**	**Romance**
1	Danish	Bulgarian	Catalan
2	Dutch	Czech	Italian
3	English	Slovene	Romanian
4	German	Russian	Spanish
5	Swedish		
SUM	5	4	4

Table 8.14: Experiment 2.1: The combination of languages within the 3 language groups including two additional languages (underlined): German and English (13 languages in total).

to the Romance group, which increases the total number of languages to 14 (see Table 8.15).

	Germanic	Slavic	Romance
1	Danish	Bulgarian	Catalan
2	Dutch	Czech	Italian
3	English	Slovene	Latin
4	German	Russian	Romanian
5	Swedish		Spanish
SUM	5	4	5

Table 8.15: Experiment 2.2: The combination of languages within the 3 language groups including one additional language (underlined): Latin (14 languages in total).

procedure	F-measure	features
`QNA[mahalanobis,hierarchical,complete]`	.92308	9/21
`QNA[mahalanobis,hierarchical,ward]`	.92308	10/21
`QNA[mahalanobis,hierarchical,ward]`	.92308	11/21
`QNA[mahalanobis,hierarchical,complete]`	.92186	7/21
`AVG over non-random approaches`	.9228	
`random baseline II`	.54203	known partition
`random baseline I`	.53414	equi-partition

Table 8.16: F-measures of classifying 13 languages into 3 genetic groups by means of QNA (Experiment 2.1).

8.5.1 Discussion on Experiment 2.1

The results of the classification experiment 2.1 are shown in Table 8.16. We see that the F-measure drops to a maximum of .92308 when we increase the total number of languages. The random baselines, which are also slightly lower than in the first experiment, are definitely surpassed. However, this combination of 13 languages does not produce a perfect classification of 1.0. We assume that biases occur due to English. This is confirmed by the values of network indices from Table 7.2. The values of lcc, Cp, C_A or ϵ for English differ more strongly compared to those of the other languages in the Germanic group. Of course, English is not a "typical" Germanic language. However, the cause for the deviation of the values for English is to be sought rather in the specifics of the underlying treebank. When we look more closely at the vertices of the English GSDN we see that 5347 vertices are very infrequent words (abbreviations, initials, etc.) and numbers (1.3, 0.342, etc.) or number-like words (11-th, 98-pound, etc.). This means, many vertices occur only once in the treebank. If such vertices are linked to a low-frequency word they are probably not part of the largest connected component. This does't hold for all of the 5347 vertices, since they can also be connected to high-frequency words. However, the illustrated specifics may explain the high number of different components and low lcc and Cp.

Despite the above observation on the large number of low-frequency words, the high number of edges compared to the number of vertices (and the high value of

procedure	F-measure	features
QNA[mahalanobis,hierarchical,complete]	.92857	12/21
QNA[mahalanobis,hierarchical,ward]	.92672	7/21
QNA[mahalanobis,hierarchical,ward]	.86349	11/21
QNA[mahalanobis,hierarchical,ward]	.63525	12/21
AVG over non-random approaches	.8385	
random baseline II	.52399	known partition
random baseline I	.52833	equi-partition

Table 8.17: F-measures of classifying 14 languages into 3 genetic groups by means of QNA (Experiment 2.2).

	Spanish	Romanian	Italian	Catalan	Latin
Spanish	0	0.0270	0.0020	0.0264	0.0009
Romanian		0	0.0000	<u>0.0816</u>	0.0000
Italian			0	0.0034	0.0000
Catalan				0	<u>0.1859</u>
Latin					0

Table 8.18: p-values of the Pearson correlation coefficient computed for the Romance group. All correlations are positive, however, not all are significant. The insignificant correlations (i.e., $p > 0.05$) are underlined. We display only one half of the matrix since the correlations are pairwise symmetric.

ϵ respectively) in the English GSDN can still reflect the typological properties of English. English has also a very low L, comparable to Japanese. We can conclude, on the one hand, the network consists of many components, on the other hand, short cuts or grammatical words ensure a high connectivity, which is generally characteristic for an analytic language. All in all, the English GSDN is definitely not similar to the other Germanic GSDNs which is due to treebank specifics as well as to the language's properties.

8.5.2 Discussion on Experiment 2.2

When examining Table 8.17 we see that an increase of languages in a group does not necessarily result in a drop of F-measure. The experiment with 14 languages achieves a higher F-measure than the experiment with 13 languages (.92308 vs. .92857). However, running Experiment 2.2 ten times we obtained only one combination of 12 features that achieves the best result. In Experiment 2.1 we got three combinations achieving the same best-off-value with 9, 10 or 11 features. This results in a better average-value for Experiment 2.1 obtained from repeating the experiments ten times each. But all in all, the differences between the best F-measure values are rather small so that the increase of the number of languages by one does not influence the result negatively. This can be due to the fact that Latin increases the overall similarity within the Romance group and makes this group more distinctive from others.

The values of the indices for Latin (Table 7.2) do not stand out from the rest of the Romance group. Latin is a rather small treebank, such as Romanian and Italian. The three treebanks are comparable in order and size. Remarkably, Italian and Latin have nearly the same values of DC, γ and $\gamma_{C(k)}$ whereby Romanian has different values for these coefficients. We computed the Pearson correlation coefficient for the Romance group in order to look more closely at the overall correlations among these languages. Although all languages within the group correlate positively according to the 21 network indices, not all correlations are significant.[11] Catalan does not correlate significantly with Romanian and Latin which is plausible from the typological point of view. Latin, Italian and Romanian exhibit perfect correlations but Spanish is also significantly correlated to the three languages. The last fact lets assume that not solely the GSDNs' order and size are responsible for the tight relations such as in the case of Romanian, Italian and Latin.

The fact that Latin and Italian have almost the same values of DC, γ and $\gamma_{C(k)}$ can be an indicator for their genealogical similarity. Italian grammar differs much from Latin. But on the lexical level there are many overlaps.

> "Das Italienische gehört zu den romanischen Sprachen, die ihrerseits als natürliche Weiterentwicklungen aus dem Lateinischen hervorgegangen sind." (*Roelcke*, 2003, 360)
> 'Italian is one of the Romance languages that can be treated as a natural further development of Latin.'[12]

Italian has preserved much of the Latin vocabulary, though not always with the same meaning as in Latin. Concerning GSDNs, DC and γ reflect the organizational principles of vocabulary (i.e., degrees), whereby $\gamma_{C(k)}$ shows the distribution of clusters and indirectly the organization of vocabulary. We can only speculate that Italian has preserved similar organizational principles as Latin, which come to the fore when looking at particular features of GSDNs. This is an interesting observation, which could mean that although Italian and Latin grammars differ significantly, some network characteristics allow to uncover their genealogical relationships. Moreover, two languages may be dissimilar on the level of grammar but still be similar on the network level, through which hidden similarities become apparent.

8.6 Experiment 3: Increasing the Number of Language Families

In this section we test the performance of QNA on 17 languages. We enlarge the number of language families by adding the Japonic (Japanese), Turkic (Turkish) and

[11] Note that we computed the Pearson correlation coefficient considering 21 features for all languages. Here too, all the correlations are positive. This fact reflects the unique universal structure of GSDNs as already observed by *Ferrer i Cancho et al.* (2004, 2007). However, some correlations are only weak and not significant, reflecting slight differences in the organization of treebanks (e.g., Turkish has the least number of significant correlations to other languages).

[12] Translation of the author of this dissertation.

procedure	F-measure	features
QNA[mahalanobis,hierarchical,complete]	.9043	7/21
QNA[mahalanobis,hierarchical,complete]	.88889	6/21
QNA[mahalanobis,hierarchical,ward]	.88889	7/21
QNA[mahalanobis,hierarchical,complete]	.87582	13/21
AVG over non-random approaches	.8893	
random baseline II	.47355	known partition
random baseline I	.48178	equi-partition

Table 8.19: F-measures of classifying 17 languages into 6 genetic groups by means of QNA (Experiment 3).

Hellenic (Greek) language families with one language each. This is not the optimal solution, since clustering with groups containing only one example are prone to errors. Unfortunately, due to the lack of data we are not able to provide any more examples of each group. However, we test the overall performance of QNA when classifying six language families. The results are discussed especially with respect to the problematic GSDNs resulting from differently structured treebanks (i.e., the Turkish treebank).

8.6.1 Discussion on Experiment 3

The results of Experiment 3 show a slight decrease in F-measure when adding three languages and three language families. The decrease is however not significant. Interestingly, the average value of Experiment 3 is higher than the average value of Experiment 2.2. The results of Experiment 3 are stable at ranging around 0.88.[13] The random baselines are clearly surpassed.

In addition, we present the dendrogram visualizing languages organized according to the best-off feature combination producing an F-measure of .9043. Figure 8.5 shows that all Germanic languages are assigned correctly to the same cluster. Further, Turkish and Japanese result in two separate clusters each just in line with their genealogical classification. Two ancient languages, Latin and Greek, share one cluster (together with Slovene, which is more similar to Greek than to the Slavic cluster). The similarities of Latin and Greek have presumably typological causes - both are, for example, highly inflectional, both have free word order. Additionally, classical Latin was strongly influenced by the Ancient Greek tradition. Of course the similarities could also have arisen from the similar annotation schemes applied for annotating the two treebanks. However, German and Danish, for instance, which are related genealogically are also similar in terms of GSDNs in spite of different annotation formalisms and even different dependency grammars used.

To summarize, we assume that the similarity between related languages can increase when the treebanks have been constructed following the same annotation principles. But the same dependency formalism does not necessarily raise the similarity of the GSDNs if two languages are unrelated to each other. For example, Italian and Danish treebanks were both constructed using Word Grammar (*Hudson*, 1984). Yet;

[13] We repeated the experiment ten times as in the case of the previous experiments.

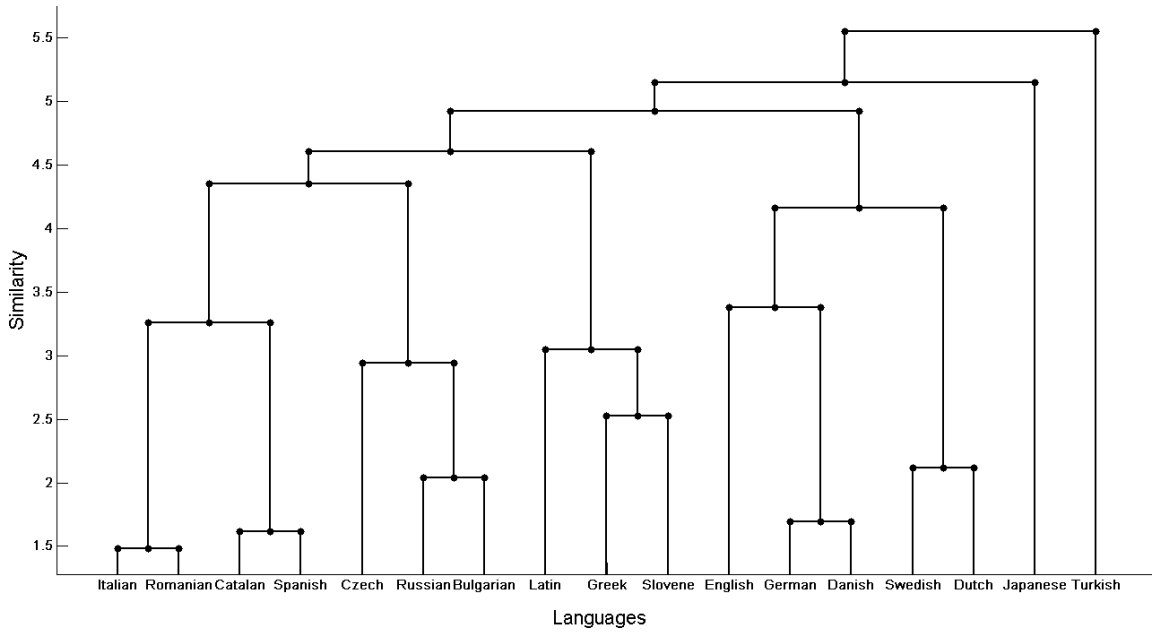

Figure 8.5: The similarity tree of languages generated by the best feature combination of QNA for 17 languages and 6 language families (Experiment 3).

the two languages are completely dissimilar in terms of their GSDNs.

8.6.2 Comparing QNA to the study of *Liu and Xu* (2011)

In this section we take a more detailed look at the study of *Liu and Xu* (2011), which is strongly related to QNA. *Liu and Xu* (2011) cluster 15 word form and lemma networks constructed as proposed in *Ferrer i Cancho et al.* (2004) based on selected network characteristics. The results fit the typological classifications of the underlying languages, that is, they are "as precise as achieved by contemporary word order typology" (*Liu and Xu*, 2011, 28005-1). The study shows that word form networks are in general more informative for classifications than lemma networks since morphological variation is also considered when using word forms. The authors suggest using degrees of function words as indicators of the degree of synthesis of a language. That is, the higher the degrees of function words are, the more synthetic the language is (*Liu and Xu*, 2011, 28005-4). All in all, the described study provides evidence for the benefit of network indices for typological research.

The indices computed in *Liu and Xu* (2011) represent a subset[14] of the indices described here. Since some treebanks used here and in *Liu and Xu* (2011) are the same, we can compare the values of the coefficients to the outcomes of QNA. Table 8.20 taken from *Liu and Xu* (2011) illustrates the values of the features. Only the first row for each language that represents the values obtained from the word form network is relevant for out purpose.

[14]The indices are: E, N, $\langle k \rangle = \epsilon$, C_{ws}, L, D, γ, $\bar{R}^2$ and NC (i.e., the network centralization).

First of all, the study operates on parts of the treebanks of about the same size in order to achieve balanced data. This results in a smaller number of vertices and edges than in our study. It makes sense to deal with balanced data, however, fluctuations of sampling could bias the indices (i.e., taking another sample may lead to different values) especially if the size of the sample is small. These fluctuations decrease when considering networks of high order and size. This is due to the small-world structure of GSDNs - when the main component of the graph is formed, adding new vertices does not change the entire structure significantly. Thus, the indices calculated for the GSDN of the whole treebank are presumably more precise. But of course, in order to achieve estimate of at which point the core structure of the small-world network is formed and small additions do not impact it any more, further analyses have to be performed.

Additionally, some indices may be biased by the order of the GSDN, others not. For example, comparing the coefficients of determination of the power-law fit of degrees obtained from the same treebanks, smaller networks (see R^2 in Table 8.20) result in worse fits (i.e., $R^2 \sim 0.7$) than larger ones (i.e., $\bar{R}^2 \sim 0.9$ in Table 7.2). The geodesic distance, in turn, is not influenced by the network order. The results for smaller and larger networks are comparable. Clustering values and average degrees are not comparable at all, which shows a dependence on the sample size. We assume that in the case of the power-law fit, the small size of the sample plays the crucial role. Treebanks of different size, which are all larger than the samples considered in *Liu and Xu* (2011), result in good fits. This can indicate that treebanks considered as a whole were large enough to reach the critical point at which the distribution does not change much any more.

Average degree is defined as $\epsilon(G) = \frac{edges}{vertices}$ in Equation 7.1. For GSDNs the number of vertices is always smaller than the number of edges ($edges > vertices$), thus, ϵ is a decreasing function. In the case of an analytic language, the number of edges will grow faster, than the number of vertices resulting in a more rapid decrease of $\epsilon(G)$ for larger graphs. In the case of synthetic languages, the difference between both values will be smaller and the function will decrease more slowly. But in any event, the dependence on size is considerable. That is, two bias factors - size/order and analyticity/synthesis - make GSDNs of different order and size incomparable to each other in terms of $\epsilon(G)$.

The results of *Liu and Xu* (2011) are presumably more precise for the particular sample size. However, if the structure of the GSDN of the particular order is still not stable, the results will vary strongly when examining comparable samples of a larger size. In our case, when considering the whole treebank (which is also not the best solution) we can expect the variations to be less significant since the entire core structure of the small-world graph is already formed. Here again, additional tests on the empirical behavior of $\epsilon(G)$ (and other coefficients) are required to estimate the point at which the coefficient stabilizes.

The values of the clustering coefficient reported in *Liu and Xu* (2011) are the values of C_{ws}. As outlined in Section 7.3, C_{ws} approaches 1 for large N. That is, the values obtained in *Liu and Xu* (2011) and in this thesis are not comparable.

Of course, the number of clusters also depends on the overall connectivity of the GSDN and on the analyticity/synthesis of the language. In the case of small-world like GSDNs, we expect that, although the coefficient approaches one for large N the average cluster value does not change much when considering large treebanks, since the vertices added are mostly low degree vertices featuring a small average cluster value. If similar cluster values for vertices are added, the graph average cluster value does not change much.

At least, the diameter is an index that increases more rapidly for synthetic languages (such as Czech) than for analytic when increasing N. In synthetic languages the probability of new word forms is higher, so that rare words are more likely to be attached to the GSDN than in analytic languages. This explains why Catalan has the identical diameter in *Liu and Xu* (2011) and in our study while Czech has a significantly larger diameter in our study than in *Liu and Xu* (2011) (cf. 16 vs. 10).

8.7 Summary

The comparison of our results with the study of *Liu and Xu* (2011) in the last section has shown that the analyses of network indices in typological studies are far from being complete. Further investigations are required in order to eliminate the possible factors biasing the outcomes. Such factors are, for example, the selection of sample size with respect to the specifics of small-world network structure of GSDNs. Some coefficients are less dependent on the sample size than others, and further studies are needed to elucidate these effects and their impact on the resulting values. Nevertheless, the small-world structure of GSDNs provides smaller biases for larger networks than for smaller ones. This is confirmed by the reliable results obtained when using the combined indices in the genealogical classification.

	E	N	$\langle k \rangle$	C	L	NC	D	γ	R^2
cat	30944	8906	6.816	0.129	3.234	0.235	9	1.165	0.703
	27484	6089	8.725	0.236	2.875	0.366	8	1.117	0.738
cze	27447	10950	4.945	0.088	3.64	0.145	10	1.254	0.692
	23527	6070	7.534	0.157	3.24	0.2	8	1.247	0.764
dut	28873	9025	6.322	0.185	3.155	0.175	8	1.085	0.703
	26495	7457	6.966	0.233	3.016	0.201	8	1.068	0.685
ell	27942	9229	5.968	0.114	3.445	0.227	11	1.226	0.722
	22660	5182	8.485	0.237	2.923	0.386	8	1.195	0.757
fre	33169	8439	7.678	0.121	3.188	0.231	9	1.173	0.717
	27837	5939	8.971	0.195	2.913	0.38	8	1.154	0.747
grc	23798	8870	5.291	0.089	3.638	0.146	11	1.343	0.746
	17984	3682	9.389	0.187	3.105	0.231	7	1.214	0.812
eus	27895	10561	5.207	0.115	3.571	0.213	13	1.334	0.75
	21883	5124	8.233	0.242	3.054	0.295	9	1.198	0.795
hun	33146	13075	5.055	0.029	3.938	0.155	11	1.353	0.734
	28975	8607	6.672	0.081	3.473	0.199	9	1.379	0.769
ita	32329	9051	7.059	0.126	3.243	0.194	8	1.185	0.701
	27484	6089	8.725	0.236	2.875	0.366	8	1.117	0.738
lat	28945	11571	4.91	0.107	3.598	0.196	11	1.266	0.721
	23848	5305	8.644	0.191	3.114	0.265	8	1.239	0.804
por	29396	8855	6.444	0.207	3.123	0.312	8	1.125	0.685
	25509	6303	7.792	0.31	2.89	0.382	8	1.12	0.716
rus	42382	16543	5.088	0.091	3.55	0.176	12	1.203	0.696
	37309	8992	8.141	0.164	3.134	0.246	10	1.249	0.745
slv	19241	7128	5.309	0.125	3.473	0.171	9	1.164	0.700
	15832	4004	7.65	0.228	2.992	0.358	7	1.171	0.759
spa	25254	7939	6.209	0.181	3.146	0.271	9	1.108	0.688
	22180	5815	7.32	0.272	2.95	0.326	8	1.101	0.716
tur	26421	11969	4.25	0.205	2.958	0.514	10	1.161	0.616
	16296	3995	7.558	0.287	2.721	0.578	8	1.229	0.773

Table 8.20: Network indices computed in *Liu and Xu* (2011) for 15 GSDNs.

CHAPTER IX

Summary and Conclusion

In this thesis we presented and discussed various approaches to enhance typological research by network modeling. These approaches aim at a *holistic* typology, allowing to make statements about a language as a whole based on different linguistic levels represented as a network. It turned out that areas such as language classification, identification, reconstruction, typology, etc. can benefit from the application of network models to language. In particular, network models analyzed in this thesis are applicable on the levels of morphology (Chapters III-IV), phonology (Chapter V) and syntax (Chapter VI-VIII).

In Chapter III, we presented a system modeling the learning of derivation morphology. The objective of this study was to investigate language change processes in a multi-agent simulation. We varied the input language (natural vs. random) and looked whether change processes are influenced by it. We found out that the structural status of the input words influences the newly emergent language. Moreover, natural languages differ in their ability to persist in generations, at least on the level of derivation morphology that we were concerned with.

In particular, we compared two languages, English and German, in terms of the amount of productive suffixes used in word formation. Although, substantial differences were observed, the evolutionary game dynamics let the agents converge on a common language. The communication was successful in all cases, even when the input language used by the adult agent was completely irregular (random). The results have shown that if the input language does not exhibit a clear structuring, children converge in the long run on common suffixes (due to the communication dynamics). However, the resulting language can be completely different from the language of the first generation agents (i.e., adults). The explicit learning (i.e., the first stage of the game) supported by the adult did not play a crucial role for the emergence of a structured input. It rather reinforced the children to learn correct word to word class relations. To a large extent, the structure of the lexical input determined the persistence of the given language. If no structural regularities were induced through the language of the adult, the outcome-language was less stable, and in most cases, the children had to negotiate other rules facilitating the communication. Here typological differences between English and German became apparent. German, which uses more suffixes in word formation, persisted in generations rather than English,

which features less structure within the word.

In Chapter IV, we extracted the so called *Morphological Derivation Networks* (MDNs) from the output of the morphological derivation game. The goal was to analyze the properties of the languages from the network perspective. In analyzing the MDNs, we could distinguish between language networks (English and German) and different random networks by means of their topological characteristics and their entropy. Language MDNs differed strongly from the random networks but also from each other revealing the typological peculiarities of English and German. For instance, the high number of self-loops in the English MDN indicates the high use of the same word forms as different parts of speech in English. This is trivial, of course, since this fact is already known for English. However, the network perspective allows us to make many such observations at once and in doing so to study the relations between particular features in language. This possibility can contribute to a better understanding of the language mechanisms as a whole, which is not easy to achieve with standard methods.

In Chapter V we presented an approach to automatically classify languages by means of phonological information obtained from the phonological database UPSID. In addition, we presented a network model that allows varying the similarity threshold to examine phonological similarities among languages and to relate them to their genealogical or typological relationships. We have found out that some language families have preserved a high phonological similarity within the family, whereby other language families exhibit a high inner-family variability (i.e., Indo-European). However, the majority of similarities among language sub-groups within single families is high, even for Indo-European. The findings indicate that changes languages undergo in time do not completely eliminate their relationship to the language family as well as to closely related languages.

In Chapters VI-VIII we presented a network approach on the level of syntax. In Chapter VI we introduced the dependency treebanks used to construct the *Global Syntactic Dependency Networks* (GSDNs). We presented 17 dependency treebanks and their qualitative and quantitative characteristics. Issues related to the specifics of dependency theories were discussed especially with respect to our data. The treebanks were originally constructed following different theoretical and language specific guidelines. Unifications on the level of punctuation (special characters) and on the level of representation formats were performed in order to achieve an approximatively optimal comparability of the data. Finally, the procedure of creating a GSDN was described.

In Chapter VII, we discussed a range of indices that are used to characterize GSDNs (and networks in general). We presented the results of computing the indices for our GSDNs and analyzed the findings. The objective of this work was to understand the impact of the indices for typological and genealogical research. It turned out that some coefficients are influenced by the specifics of annotation, especially by the artificiality of data. We also detected some unexpected correlations between the indices by performing an analysis of correlation. Features that are correlated with many other features are presumably less informative for classification, because they

are redundant. It was sufficient to consider one of the correlated features, and in so doing, reducing the classification's total feature space.

Finally in Chapter VIII, we presented the results of the genealogical classification of languages by means of the topological indices discussed in the previous chapter. The overall results are encouraging, showing that genealogical relations are traced back by examining the structure of the GSDN. The comparison of our results with the study of *Liu and Xu* (2011) in the last section of the chapter proved that the analysis of network indices in typological studies being far from being complete. Further investigations are required in order to eliminate the possible factors biasing the outcomes. Such factors are, for example, the selection of sample size with respect to the specifics of small-world network structure of GSDNs. Some coefficients turned out to be less dependent on the sample size than others, and further studies are needed to elucidate these effects and their impact on the resulting values. Nevertheless, the small-world structure of GSDNs produces smaller biases for larger networks than for smaller ones. This is confirmed by the good results obtained when using the combined indices in classification.

Our overall results have shown that although related languages diverge over time showing not much of the family-similarity, inner structuring on morphological, phonological or syntactic levels can nevertheless recover this similarity. Furthermore, dependencies and relations between particular language characteristics can be studied using networks. Thus, the instruments proposed here can enhance typological, genealogical or areal research by delivering additional insights into the structure of languages.

So far, we presented network approaches attributed to a particular linguistic level (morphology, phonology, syntax). However, networks of different levels can be combined in order to study the interplay of single linguistic levels and to understand their synergy as postulated by *Köhler* (1986). Holistic typology as well as explanations of change processes languages undergo and related issues can be addressed by means of combined linguistic networks. In future work, we aim to extend the work presented in this thesis by combining different kinds of linguistic networks.

APPENDICES

APPENDIX A

The Suffix Induction Algorithm

```
for all word classes c do
  for all word ⟨w, c⟩ do
    { 1. filtering step}
    for all s ∈ w {suffix} do
      N(s) = number of words of class c containing s
      if N(s) ∈ L(c) ≥ 20% of |L(c)| then
        group_list_w.push_back(s)
      end if
    end for
    { 2. filtering step}
    sim_val = 0.1
    for all group_list_w do
      s1, s2 ∈ group_list_w, s_k ≠ s_l, s1 ∈ s2,
      Sim(s1, s2) {compares the frequencies of s1 and s2 in L(c)}
      if Sim(s_k,s_l) ≤ sim_val then
        delete s1 {'ich', 'ch' → the shorter, i.e., 'ch', is removed}
      end if
    end for
    {3. filtering step}
    sim_val = 0.0000001
    for all group_list_w do
      list_from_all_words.push_back(group_list_w)
    end for
    for all s1, s2 ∈ list_from_all_words, s1 ≠ s2, s1 ∈ s2, do
      Sim(s1, s2)
      if Sim(s1,s2) ≤ sim_val then
        delete s1 {'ich', 'ch' → 'ch' is removed}
      end if
    end for{4. filtering step} {create suffix trees for each s1}
    for all s1, s2 ∈ list_from_all_words, s1 ≠ s2 do
      if s1 ∈ s2 then
        suffixtree(s1).add(s2)
      end if
    end for
    for all s ∈ list_from_all_words do
      for all s1, s2 ∈ suffixtree (s), s1 ≠ s2 do
        if (s1 ∩ s2) → s then
          mark(s1)
          mark(s2)
        end if
      end for
    end for
    for all s ∈ list_from_all_words do
      Pr(s) = |A(s) \ M(s)|
      {A(s) : suffix tree of s, M(s) : marked suffixes of s}
    end for
  end for
end for
```

Figure A.1: Suffix induction from single words in four filtering steps (see *Pustylnikov* (2010)). For each word and each word class: 1) collect all suffixes of a word, 2) filter out suffixes of a similar frequency (according to a similarity threshold) and remove the shorter of the mutually inclusive suffixes. 3) do the same as in 2) for all suffixes of all words with a reduced similarity threshold. 4) construct a suffix tree for each suffix retained after filtering 1-3. Remove all marked suffixes from each suffix tree that contain a larger common part than the considered suffix, i.e., remove 'lich' and 'klich' from the tree of 'ch', since they are already present in 'lich'. $Pr(s)$ is the number the different suffixes, the suffix s is present in. Rank all suffixes according to their $Pr(s)$ values.

BIBLIOGRAPHY

BIBLIOGRAPHY

Abramov, O., and T. Lokot (2011), *Typology by Means of Language Networks: Applying Information Theoretic Measures to Morphological Derivation Networks*, chap. 11, Springer.

Abramov, O., and A. Mehler (2011), Automatic language classification by means of syntactic dependency networks, *Journal of Quantitative Linguistics*, accepted.

Ahmed, B., S.-H. Cha, and C. Tappert (2004), Language identification from text using n-gram based cumulative frequency addition, in *Pace CSIS Research Day.*

Alava, M. J., and S. N. Dorogovtsev (2004), Preferential compactness of networks, *ArXiv Condensed Matter e-prints.*

Altmann, G., and V. Altmann (2005), Erlkönig und mathematik, http://ubt.opus.hbz-nrw.de/volltexte/2005/325/.

Altmann, G., and W. Lehfeldt (1973), *Allgemeine Sprachtypologie*, Wilhelm Fink.

Andreev, N. D. (1967), *Statistiko-kombinatornye metody v teoretičeskom i prikladnom jazykovedenii*, Leningrad.

Anttila, R. (1972), *An introduction to historical and comparative linguistics*, The Macmillan Company, New York.

Baayen, H. (1991), Quantitative Aspects of Morphological Productivity, in *Yearbook of Morphology*, edited by J. M. Geert Booij, pp. 109–149, Kluwer, Dordrecht - Boston - London.

Baayen, H. (1992), On frequency, transparency, and productivity, *Yearbook of Morphology 1992*, pp. 181–208.

Bakker, D., et al. (2009), Adding typology to lexicostatistics: a combined approach to language classification, *Linguistic Typology*, *13*, 167–179.

Bamman, D., and G. Crane (2006), The Design and Use of a Latin Dependency Treebank, in *Proc. of the Fifth International Workshop on Treebanks and Linguistic Theories (TLT 2006)*, pp. 67–78.

Barabási, A.-L., and R. Albert (1999), Emergence of scaling in random networks, *Science*, *286*, 509–512.

Barrat, A., M. Barthélemy, and A. Vespignani (2008), *Dynamical processes on complex networks*, Cambridge University Press.

Batagelj, V., D. Kerzic, and T. Pisanski (1992), Automatic clustering of languages, *Computational Linguistics*, *18*(3).

Bender, M. L. (1997), *The Nilo-Saharan languages. A comparative essay*, LINCOM Europa, München.

Bertinetto, P. M., and S. Noccetti (2006), Prolegomena to ATAM acquisition. Theoretical premises and corpus labeling, *Quaderni del Laboratorio di Linguistica della SNS n.6 ns.*

Blanchard, P., F. Petroni, M. Serva, and D. Volchenkov (2009), Networking phylogeny for indo-european and austronesian languages, Nature Precedings [http://precedings.nature.com/oai2] (United States).

Bloomfield, L. (1933), *Language*, Henry Holt and Co.

Boguslavsky, I., I. Chardin, S. Grigorieva, N. Grigoriev, L. Iomdin, L. Kreidlin, and N. Frid (2002), Development of a dependency treebank for russian and its possible applications in NLP, in *Proc of LREC 2002*.

Bollobás, B., and O. M. Riordan (2003), Mathematical results on scale-free random graphs, in *Handbook of Graphs and Networks. From the Genome to the Internet*, edited by S. Bornholdt and H. G. Schuster, pp. 1–34, Wiley-VCH, Weinheim.

Bosco, C., V. Lombardo, D. Vassallo, and L. Lesmo (2000), Building a treebank for Italian: a data-driven annotation schema, in *Proc. of LREC 2000*.

Botafogo, R. A., E. Rivlin, and B. Shneiderman (1992), Structural analysis of hypertexts: Identifying hierarchies and useful metrics, *ACM Transactions on Information Systems*, *10*(2), 142–180.

Brandes, U. (2001), A faster algorithm for betweenness centrality, *Journal of Mathematical Sociology*, *25*, 163–177.

Brants, S., S. Dipper, S. Hansen, W. Lezius, and G. Smith (2002), The TIGER treebank, in *"Proceedings of the Workshop on Treebanks and Linguistic Theories"*, Sozopol.

Bryant, D., F. Filimon, and R. Gray (2005), *Untangling our past: Languages, trees, splits and networks*, pp. 67–84, UCL Press, London.

Bybee, J. L. (1988), *Morphology as lexical organization*, chap. 7, pp. 119–141, Academic Press.

Caldarelli, G., and A. Vespignani (2007), *Large scale structure and dynamics of complex networks. From information technology to finance and natural science*, World Scientific Publishing.

Campbell, L. (2006), Areal linguistics: the problem to the answer, in *Language contact and areal linguistics.*, edited by N. V. April McMahon and Y. Matras, Houndmills, Basingstoke.

Cavnar, W. B., and J. M. Trenkle (1994), N-gram-based text categorization, in *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pp. 161–175, Las Vegas, US.

Chomsky, N. (1957), *Syntactic Structures*, The Hague: Mouton.

Chomsky, N. (1965), *Aspects of the Theory of Syntax*, MIT Press.

Choudhury, M., and A. Mukherjee (2009), The structure and dynamics of linguistic networks, in *Dynamics On and Of Complex Networks*, edited by N. Ganguly, A. Deutsch, and A. Mukherjee, Modeling and Simulation in Science, Engineering and Technology, pp. 145–166, Birkhäuser Boston.

Churcher, G., J. Hayes, S. Johnson, and C. Souter (1994), Bigraph and trigraph models for language identification and character recognition, in *Proc. of 1994 AISB Workshop on Computational Linguistics for Speech and Handwriting Recognition*, Leeds.

Civit, M., and M. Martí (2005), Building Cast3LB: A Spanish Treebank, a Research on Language and Computation, *Springer Verlag*, pp. 549–574.

Civit, M., N. B. i, and P. Valverde (2004), CAT3LB: a Treebank for Catalan with Word Sense Annotation, in *TLT2004*, pp. 27–38, Tubingen University.

Clahsen, H., I. Sonnenstuhl, and J. P. Blevins (2003), Derivational morphology in the german mental lexicon: a dual mechanism account, in *In: H. Baayen & R. Schreuder (Eds.), Morphological structure in language processing, Mouton de Gruyter*, pp. 125–155, 2006.

Combrinck, H., and E. Botha (1995), Text-based automatic language identification, in *Proceedings of the Sixth Annual South African Workshop on Pattern Recognition*, Rand Afrikaans University, Gauteng, South Africa.

Daumé III, H. (2009), Non-parametric Bayesian model areal linguistics, in *North American Chapter of the Association for Computational Linguistics (NAACL)*, Boulder, CO.

De Boer, B. (2001), *The Origins of Vowel Systems*, Oxford University Press.

Debusmann, R. (2000), An introduction to dependency grammar, Hausarbeit.

Dehmer, M. (2008), Information processing in complex networks: Graph entropy and information functionals, *Applied Mathematics and Computation*, *201*, 82–94.

Dehmer, M., K. Varmuza, S. Borgert, and F. Emmert-Streib (2009), On entropy-based molecular descriptors: statistical analysis of real and synthetic chemical structures, *Journal of chemical information and modeling*, *49*(7), 1655–1663.

den Boogaard, U. (1975), *Woordfrequenties in geschreven en gesproken Nederlands*, Oosterhoek, Scheltema & Holkema, Utrecht.

Diestel, R. (2006), *Graphentheorie*, 3 ed., Springer.

Dressler, W. U., and A. Karpf (1995), The theoretical relevance of pre- and protomorpholgy in language acquisition, *Yearbook of Morphology 1994*, pp. 99–122.

Dunning, T. (1994), Statistical identification of language, *Tech. Rep. MCCS-94-273*, Computing Research Lab (CRL), New Mexico State University.

Džeroski, S., T. Erjavec, N. Ledinek, P. Pajas, Z. Žabokrtský, and A. Žele (2006), Towards a Slovene dependency treebank, in *Proc. of LREC 2006*.

Ehret, C. (2001), *A historical-comparative reconstruction of Nilo-Saharan*, Köppe, Köln.

Erdős, P., and A. Rényi (1959), On random graphs, *Publicationes Mathematicae*, *6*, 290–297.

Eryiğit, G., J. Nivre, and K. Oflazer (2008), Dependency parsing of turkish, *Comput. Linguist.*, *34*, 627–627.

Feldman, R., and J. Sanger (2007), *The Text Mining Handbook. Advanced Approaches in Analyzing Unstructured Data*, Cambridge University Press, Cambridge.

Ferrer i Cancho, R. (2003), Language: universals, principles and origins, Ph.D. thesis, Universitat Politècnica de Catalunya, Barcelona.

Ferrer i Cancho, R., and R. V. Sole (2001), The small world of human language, *Proceedings of The Royal Society of London. Series B, Biological Sciences*, *268*(1482), 2261–2265.

Ferrer i Cancho, R., and R. V. Solé (2003), Least effort and the origins of scaling in human language, *Proc. of the National Academy of Sciences USA*, *100*, 788–791.

Ferrer i Cancho, R., R. V. Solé, and R. Köhler (2004), Patterns in syntactic dependency networks, *Physical Review E*, *69*, 051,915.

Ferrer i Cancho, R., A. Mehler, O. Pustylnikov, and A. Díaz-Guilera (2007), Correlations in the organization of large-scale syntactic dependency networks, in *TextGraphs-2: Graph-Based Algorithms for Natural Language Processing*, pp. 65–72.

Ford, P. D., and T. Voegtlin (2003), Learning word meaning and grammatical constructions from narrated video events, in *Proceedings of the HLT-NAACL 2003 workshop on Learning word meaning from non-linguistic data*, pp. 38–45, Association for Computational Linguistics, Morristown, NJ, USA.

Freeman, L. C. (1978-1979), Centrality in social networks conceptual clarification, *Social Networks*, *1*(3), 215 – 239.

Goldsmith, J. (2001), Unsupervised learning of the morphology of a natural language., *Computational Linguistics*, *27*(2), 153–198.

Gong, T., J. W. Minett, and W. S.-Y. Wang (2009), A simulation study on word order bias, *Interaction Studies*, *10*(1), 51–75.

Greenberg, J. H. (1966), Some universals of grammar with particular reference to the order of maningful elements, in *Universals of Language*, edited by J. Greenberg, MIT Press, Cambridge.

Grefenstette, G. (1995), Comparing two language identification schemes, in *4rd Int. Conference on the Statistical Analysis of Textual Data*, Rome, Italy.

Hage, P., and F. Harary (1995), Eccentricity and centrality in networks, *Social Networks*, *17*, 57–63.

Hajič, J. (1998), "Building a Syntactically Annotated Corpus: The PragueDependency Treebank", in *Issues of Valency and Meaning. Studies in Honour of JarmilaPanevová*, edited by E. Hajičová, pp. 106–132, Karolinum, Charles University Press, Prague, Czech Republic.

Harary, F. (1959), Status and contrastatus, *Sociometry*, *22*(1), 23–43.

Harris, Z. (1967), Morpheme boundaries within words: Report on a computer test, in *Transformations and Discourse Analysis Papers 73*, Department of Linguistics, University of Pennsylvania.

Haspelmath, M., M. Dryer, D. Gil, and B. Comrie (2005), *The World Atlas of Language Structures*, Oxford University Press, Oxford.

Havelka, J. (2007), Beyond projectivity: Multilingual evaluation of constraints and measures on non-projective structures, in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 608–615, Association for Computational Linguistics, Prague, Czech Republic.

Hawkins, J. A. (1983), *Word Order Universals*, Academic Press., London.

Hinrichs, E. W., J. Bartels, Y. Kawata, V. Kordoni, and H. Telljohann (2000), The verbmobil treebanks, in *KONVENS 2000 / Sprachkommunikation, Vorträge der gemeinsamen Veranstaltung 5. Konferenz zur Verarbeitung natürlicher Sprache (KONVENS)*, pp. 107–112, VDE-Verlag GmbH, Berlin, Germany, Germany.

Holman, E. W., S. Wichmann, C. H. Brown, V. Velupillai, A. Müller, and D. Bakker (2008), Explorations in automated language classification, *Folia Linguistica*, *42*(2), 331–354.

Hotho, A., A. Nürnberger, and G. Paaß (2005), A Brief Survey of Text Mining, *Journal for Language Technology and Computational Linguistics (JLCL)*, *20*(1), 19–62.

Hudson, R. (1984), *Word Grammar*, Blackwell.

Hudson, R. (1994), Discontinuous phrases in dependency grammar, *UCL Working Papers in Linguistics 6*, pp. 89–120.

Jäger, G. (2006), Das A und O des Sprechens, *Gehirn & Geist*, *11*, 70–76.

Johannesson, M. (2000), Modelling asymmetric similarity with prominence, *British Journal of Mathematical and Statistical Psychology*, *53*(1), 121–139.

Kakkonen, T. (2005), Dependency Treebanks: Methods, Annotation Schemes and Tools, in *Proceedings of the 15th Nordic Conference of Computational Linguistics (NODALIDA 2005)*, pp. 94–104, Joensuu, Finland.

Kapatsinski, V. (2008), Principal components of sound systems: An exercise in multivariate statistical typology, *IULC Working Papers*, *8*.

Kaplan, J., and J. Bresnan (1982), *Lexical-Functional Grammar: A formal system for grammatical representation*, Cambridge University Press.

Kello, C. T., and B. C. Beltz (2009), Scale-free networks in phonological and orthographic wordform lexicons, *To appear in: Approaches to Phonological Complexity.*

Klein, G. G. E., and I. S. und Geoffrey Pullum (1985), *Generalized Phrase Structure Grammar*, B. Blackwell.

Klimov, G. (1980), *K vsaimootnošeniju genealogičeskoj, tipologičeskoj i areal'noj klassifikazii jasykov*, chap. 1, pp. 6–23, Nauka.

Köhler, R. (1986), *Zur linguistischen Synergetik: Struktur und Dynamik der Lexik*, Brockmeyer, Bochum.

Kondrak, G. (2002), Algorithms for language reconstruction, Ph.D. thesis, University Toronto.

Konstantinova, E. V. (2006), On some applications of information indices in chemical graph theory, in *General Theory of Information Transfer and Combinatorics*, Springer.

Konstantinova, E. V., and A. A. Paleev (1990), Sensitivity of topological indices of polycyclic graphs (Russian), *Vichisl. Systemy*, *136*, 38–48.

Krioukov, D., K. Fall, and X. Yang (2004), Compact routing on internet-like graphs, in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, p. 219.

Kromann, M. T. (2003), The danish dependency treebank and the underlying linguistic theory, in *Proc. of TLT 2003*, edited by J. Nivre and E. Hinrichs, Växjö University Press.

Kruskal, J. B., P. Black, and I. Dyen (1992), *An Indo-European Classification. A Lexicostatistical Experiment (Transactions of the American Philosophical Society)*, Amer Philosophical Society.

Lewis, D. K. (1969), *Convention*, Harvard University Press, Harvard, Mass.

Lins, R. D., and P. Gonçalves (2004), Automatic language identification of written texts, in *Proc. of the ACM SAC '04*, pp. 1128–1133, ACM, New York, NY, USA.

Liu, H. (2008), The complexity of chinese syntactic dependency networks, *Physica A 387*, pp. 3048–3058.

Liu, H., and C. Xu (2011), Can syntactic networks indicate morphological complexity of a language?, *EPL (Europhysics Letters)*, *93*(2), 28,005.

Liu, H., Y. Zhao, and W. Huang (2010), How do local syntactic structures influence global properties in language networks?, *Glottometrics*, *20*, 38–58.

Maddieson, I. (1984), *Patterns of sound*, Cambridge University Press.

Maddieson, I., and K. Precoda (1989), Updating upsid, *Journal of the Acoustical Society of America*, *86*, 19–19.

Marcus, M., B. Santorini, and M. A. Marcinkiewicz (1993), Building a large annotated corpus of English: the Penn Treebank, in *Computational Linguistics 19*.

Masayoshi, S., and T. Bynon (1995), Approaches to Language Typology. A Conspectus, in *Approaches to Language Typology*.

McNamee, P. (2005), Language identification: a solved problem suitable for undergraduate instruction, *J. Comput. Small Coll.*, *20*(3), 94–101.

Mehler, A. (2008a), Structural similarities of complex networks: A computational model by example of wiki graphs, *Applied Artificial Intelligence*, *22*, 619–683.

Mehler, A. (2008b), A short note on social-semiotic networks from the point of view of quantitative semantics, in *Proceedings of the Dagstuhl Seminar on Social Web Communities*, edited by H. Alani, S. Staab, and G. Stumme, Dagstuhl.

Mehler, A. (2009), A quantitative graph model of social ontologies by example of Wikipedia, in *Genres on the Web: Computational Models and Empirical Studies*, edited by A. Mehler, S. Sharoff, and M. Santini, pp. 291–352, Submitted to Springer, Berlin/New York.

Mehler, A., P. Geibel, and O. Pustylnikov (2007), Structural Classifiers of Text Types: Towards a Novel Model of Text Representation, *LDV Forum*, pp. 51–66.

Mehler, A., O. Pustylnikov, and N. Diewald (2010a), The geography of social ontologies: The sapir-whorf hypothesis revised, *Computer, Speech and Language. Special Issue on Network models of social and cognitive dynamics of language.*

Mehler, A., P. Weiß, P. Menke, and A. Lücking (2010b), Towards a simulation model of dialogical alignment, in *Proceedings of the 8th International Conference on the Evolution of Language (Evolang8).*

Mehler, A., T. Lokot, and O. Abramov (In preparation), Towards an adequate measure of compactness of graphs, in preparation.

Mel'čuk, I. (1988), *Dependency Syntax: Theory and Practice*, State University Press of New York.

Mengel, A., and W. Lezius (2000), An XML-based representation format for syntactically annotated corpora, in *Proceedings of the International Conference on Language Resources and Evaluation*, pp. 121–126, Athens, Greece.

Minkov, E., and W. W. Cohen (2008), Learning graph walk based similarity measures for parsed text, in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pp. 907–916, Association for Computational Linguistics, Honolulu, Hawaii.

Mukherjee, A., M. Choudhury, A. Basu, and N. Ganguly (2009), Self-organization of the sound inventories: Analysis and synthesis of the occurrence and co-occurrence networks of consonants, *Journal of Quantitative Linguistics*, *16*(2), 157–184.

National Science Foundation (2001), The National Laboratory for Applied Network Research (NLANR), `http://moat.nlanr.net/`.

Newman, M. E. J. (2003), The structure and function of complex networks, *SIAM Review*, *45*, 167–256.

Newman, M. E. J., and J. Park (2003), Why social networks are different from other types of networks, *Physical Review E*, *68*, 036,122.

Newmann, M. (2002), Assortative mixing in networks, *Physical Review Letters*, *89*, 208,701.

Nivre, J. (2005), Dependency Grammar and Dependency Parsing, *Tech. Rep. MSI report 05133*, Växjö University: School of Mathematics and Systems Engineering.

Nivre, J. (2006), *Inductive Dependency Parsing*, Springer.

Nivre, J., J. Nilsson, and J. Hall (2006), Talbanken05: A swedish treebank with phrase structure and dependency annotation, in *Proc. of LREC 2006*.

Oflazer, K., B. Say, D. Z. Hakkani-Tür, , and G. Tür (2003), *Building a Turkish treebank*, chap. 1, pp. 1–17, Kluwer Academic Publishers.

Oostdijk, N. (2000), The spoken dutch corpus: Overview and first evaluation, in *Proceedings of the Second International Conference on Language Resourcesand Evaluation (LREC)*, pp. 887–894.

Osenova, P., and K. Simov (2004), BTB-TR05: BulTreeBank Stylebook. BulTreeBank Project Technical Report Nr. 05, *Tech. rep.*, Linguistic Modelling Laboratory, Bulgarian Academy of Sciences.

Oswalt, R. L. (1970), The detection of remote linguistic relationships, *Computer Studies in the Humanities and Verbal Behavior*, *3*, 117–129.

Pastor-Satorras, R., A. Vázquez, and A. Vespignani (2001), Dynamical and correlation properties of the internet, *Physical Review Letters*, *87*, 258,701.

Pinkster, H. (1990), *Latin Syntax and Semantics*, Routledge, London.

Pollard, C., and I. A. Sag (1994), *Head-Driven Phrase Structure Grammar*.

Pollard, C. J., and I. A. Sag (1988), An information-based theory of agreement, in *Proc. of the 24th Regional Meeting of the Chicago Linguistic Society*, CLS, Chicago, Illinois.

Pustylnikov, O. (2009a), Modeling learning of derivation morphology in a multi-agent simulation, in *Proceedings of IEEE Africon 2009*, IEEE.

Pustylnikov, O. (2010), Automatic language classification by means of networks, Ph.D. thesis, Bielefeld University, In preparation.

Pustylnikov, O., and A. Mehler (2007), Structural differentiae of text types. a quantitative model, in *Proceedings of the 31st Annual Conference of the German Classification Society on Data Analysis, Machine Learning, and Applications (GfKl)*.

Pustylnikov, O., and A. Mehler (2008), Towards a uniform representation of treebanks: Providing interoperability for dependency tree data, in *Proceedings of First International Conference on Global Interoperability for Language Resources (ICGL 2008), Hong Kong SAR, January 9-11*.

Pustylnikov, O., and K. Schneider-Wiejowski (2009), Measuring morphological productivity, *Studies in Quantitative Linguistics 5: Issues in Quantitative Linguistics*, pp. 106–125.

Pustylnikov, O., A. Mehler, and R. Gleim (2008), A unified database of dependency treebanks. Integrating, quantifying & evaluating dependency data, in *Proceedings of the 6th Language Resources and Evaluation Conference (LREC 2008), Marrakech (Morocco).*

Pustylnikov, R. (2009b), Sprachsimulationssoftware für linguistische forschungszwecke, Master's thesis, FH Bielefeld University of Applied Science.

Ravasz, E., and A.-L. Barabási (2003), Hierarchical organization in complex networks, *Physical Review E*, *67*, 026,112.

Robinson, J. J. (1970), Dependency structures and transformation rules, *Language*, *46*, 259–285.

Roelcke, T. (Ed.) (2003), *Variationstypologie. Ein sprachtypologisches Handbuch der europäischen Sprachen: Ein Sprachtypologisches Handbuch Der Europaischen Sprachen*, de Gruyter.

Römer, C., and B. Matzke (2005), *Lexikologie des Deutschen*, 2 ed., Gunter Narr Verlag.

Ross, M. (1988), *Proto Oceanic and the Austronesian languages of Western Melanesia*, Canberra, A.C.T.

Sang, E. F. T. K., and S. Buchholz (2000), Introduction to the conll-2000 shared task: Chunking.

Sgall, P. (1995), *Prague School Typology*, chap. 3, pp. 49–84, Clarendon Press.

Sgall, P., E. Hajicová, J. Panevová, and J. Mey (1986), *The Meaning of the Sentence in its Semantic and Pragmatic Aspects*, Springer Verlag.

Simpson, A. P. (1999), Fundamental problems in comparative phonetics and phonology. does upsid help to solve them?, in *XIVth ICPhS*, pp. 349–352, San Francisco.

Skalička, V. (1935), *Zur ungarischen Grammatik*, Prag.

Skalička, V. (1979), *Typologische Studien*, Vieweg, Braunschweig / Wiesbaden.

Skut, W., B. Krenn, and H. Uszkoreit (1997), An annotation scheme for free word order languages, in *Proceedings of the Fifth Conference on Applied Natural Language Processing*, Washington, DC.

Smith, J. M. (1982), *Evolution and the Theory of Games*, Cambridge University Press.

Snijders, T. A. B. (1981), The degree variance: An index of graph heterogeneity, *Social Networks*, *3*(3), 163 – 174.

Soffer, S. N., and A. Vázquez (2005), Network clustering coefficient without degree-correlation biases, *Phys. Rev. E*, *71*(5), 057,101, doi:10.1103/PhysRevE.71.057101.

Sullivan, J., and A. McMahon (2010), Phonetic comparison, varieties, and networks: Swadesh's influence lives on here too, *Diachronica*, pp. 325–340.

Surdeanu, M., R. Johansson, L. Màrquez, A. Meyers, and J. Nivre (2009), 2008 conll shared task data, Linguistic Data Consortium, Philadelphia.

Swadesh, M. (1952), Lexico-statistic dating of prehistoric ethnic contacts, in *Proc. Am. Phil. Soc.*, pp. 453–463.

Takci, H., and I. Sogukpinar (2004), Centroid-based language identification using letter feature set, in *LNCS 2945*, edited by A. Gelbukh, pp. 640–648, Springer-Verlag.

Tambovtsev, Y. (2007), How can typological distances between latin and some indo-european language taxa improve its classification?, *The Prague Bulletin of Mathematical Linguistics*, (88), 73–90.

Teh, Y. W., H. Daumé, and D. Roy (2009), Bayesian agglomerative clustering with coalescents, in *Proceedings of the Conference on Neural Information Processing Systems(NIPS)*.

Teich, E. (1998), Language identification from text using n-gram based cumulative frequency addition, in *In Proc. of Dependency-based Grammars: Proceedings of the Workshop, COLING-ACL'98*.

Tesnière, L. (1959), *Eléments de syntaxe structurale*, Klincksieck, Paris.

Thomason, S. G., and T. Kaufman (1988), *Language Contact, Creolization, and Genetic Linguistics*, Univ of California Press.

Tomasello, M. (2005), *Constructing a Language : A Usage-Based Theory of Language Acquisition*, Harvard University Press.

UCLA, and D. Eisenberg (2010), The Database of Interacting Proteins (DIP), `http://dip.doe-mbi.ucla.edu/dip`.

Ukkonen, E. (1995), On-line construction of suffix trees, *Algorithmica*, *14*(3), 249–260.

van der Beek, L., G. Bouma, R. Malouf, and G. van Noord (2002), The Alpino dependency treebank, in *Computational Linguistics in the Netherlands CLIN*, Radopi.

Vogt, P. (2005), The emergence of compositional structures in perceptually grounded language games, *Artificial Intelligence*, *167*(1-2), 206–242.

von der Gabelentz, G. (1901), *Die Sprachwissenschaft. Ihre Aufgaben, Methoden und bisherigen Ergebnisse*, Chr. H. Tauchnitz, Leipzig.

Warnow, T., D. Ringe, and A. Taylor (1996), Reconstructing the evolutionary history of natural language, in *Society of Industrial and Applied Mathematics, Proc. of ACM-SIAM Symposium on Discrete Algorithms (SODA*, pp. 314–322.

Wasserman, S., and K. Faust (1999), *Social Network Analysis. Methods and Applications*, Cambridge University Press, Cambridge.

Watts, D. J., and S. H. Strogatz (1998), Collective dynamics of 'small-world' networks, *Nature*, *393*, 440–442.

Whaley, L. J. (1997), *Introduction to Typology : The Unity and Diversity of Language* , Sage Publications.

Williamson, G. (2009), Type-token ratio, `http://www.speech-therapy-information-and-resources.com/type-token-ratio.html`.

Zipf, G. K. (1932), *Selected Studies of the Principle of Relative Frequency in Language*, Harvard University Press, Cambridge (Mass.).

Zipf, G. K. (1949), *Human Behavior and the Principle of Least Effort*, Addison-Wesley.