Modeling Linguistic Typology - A Probabilistic Graphical Models Approach

Xia Lu
Emerging Technology Center Co., Ltd. Midea
robin.lu@midea.com

1 Introduction

In this paper we propose to use probabilistic graphical models (PGM) as a new theoretical and computational framework to study linguistic typology. Such models can be used by researchers within the linguistics community to study classical problems in linguistic typology such as langauge universals. Researchers from the NLP community will also find them useful for tasks like predicting typological feature values of new languages as described in (Bjerva et al., 2020).

Developed based on both probabilistic theory and graph theory probabilistic graphical models (PGM) use graph structure to represent the interactions between variables in a multidimensional distribution (Koller & Friedman, 2009). We propose the concept of a meta-language that has the universal properties of all languages in the world. This meta-language is defined by a limited set of linguistic features and the relationships among them. The graphical structure of a PGM provides an ideal representation for the meta-language. The probabilistic inference enables us to quantify the relationships among linguistic features using probability which describes the degree of confidence about the uncertain nature of typological feature correlations. Between the two types of PGMs: directed acyclic graph (Bayesian Network) and undirected acyclic graph (Markov Network) we chose the directed one. To construct a model, we learn structure and parameters from data. There are three difficulties in learning a PGM from the dataset we use which is WALS (Dryer & Haspelmath, 2013). The biggest difficulty which is also the classical problem in linguistic typology is often referred to as "geographical and genealogical biases". Specifically the language samples in the WALS database are not independent and identically distributed (i.i.d.) because languages can share the same feature values due to either genetic or areal proximity (Dryer, 1989; Croft, 2002). Here we propose two models: one is FLAT, which assumes samples are independent and identically distributed (i.i.d.); the other is UNIV, which takes care of the possible dependencies among the samples. By comparing these two models we hope to find one that is closer to the real distribution. Aside from the i.i.d. problem there are two other issues with the dataset: one is large quantity of missing values and the other is small number of instances, both of which are addressed in our algorithms.

2 Experiments

In the WALS dataset there are 192 features in total. Learning models with all these features is still ongoing. Here we use the word order domain as an example. First, we manually select the features that are well defined and most representative of the domain, reducing the original 56 features to 13.

One important part of our algorithm is model averaging or graph fusion which refers to the process of aggregating multiple graph structures and identify the features of higher confidence. Specifically for m number of datasets D_i (i=1, 2,…m), we induce a network structure G_i=G(D_i) from each dataset, and define:

$$p(f) = \frac{1}{m}\sum_{i=1}^{m} f(G_i) \qquad (1)$$

where $p(f)$ represents confidence of existence of feature f in the final structure G: if the value is close to 1 then f is present in G; if the value is close to 0 then it is not. In (Friedman et al., 1999) such features refers to edges in partially directed graphs (PDAGs) , Markov neighborhoods and orders. In this paper we use edges alone. From the linguistic perspective we can assume such measure as degree of universality: the larger the value is the more likely that this feature is universally found in all languages. One advantage of using PGMs is that it allows us to aggregate and average on the model structure level.

To learn a FLAT model, we use an iterative procedure combining imputation and both parametric and non-parametric bootstrap methods. A graphical illustration of the algorithm is shown in Figure 1. To learn a UNIV model we assume languages belongs to different groups along both genealogical and areal scales. Following the model averaging principle discussed in the previous section we assume for each group we can learn a set of structures then through aggregating and averaging we can identify the edges that have higher probabilities of being universally present across language groups. We chose the "Macroarea" classification as defined in (Dryer, 1989) which is based on geographical boundaries primarily but also taking genealogical grouping into account. To learn a UNIV model we learn local FLAT models for each area and aggregate all local models to get a final model. The structure of the UNIV model is shown in Figure 2. The confidence value is added next to each edge and also reflects in the thickness of the arrows.

We use Bayesian score to evaluate the structure and prediction accuracy to evaluate the parameters. The accuracy for one-feature prediction for one language l is defined as such:

$$acc_l^1 = \frac{number\ of\ corrected\ predicted\ values}{total\ number\ of\ features} \quad (2)$$

For a dataset with n languages the final accuracy for one-feature prediction is:

$$acc^{1f} = \frac{1}{n}\sum_{i=1}^{n} acc_i^{1f} \quad (3)$$

Besides the FLAT and UNIV models we also learnt two more models: one is COMP which is learnt using 188 instances with no missing values and another EM which is learnt from the original dataset using the SEM algorithm (Friedman, 1998). The structure score and one-feature accuracy of the four models are shown in Figure 3 and 4. For accuracy calculation we use the 188 complete cases as the test set. It is not surprising that COMP has the highest values for both while UNIV outperforms both EM and FLAT.

3 Results

In this section we discuss how to use a PGM to study linguistic typology. For better visualization we use SamIam which is a tool for modeling and reasoning with Bayesian networks. First we can exam the relationships between the typological features visually from the graph. Instead of simply stating X is correlated to Y, we see correlation between two features are relative rather than absolute: the correlation can be present or absent depending on the information we have about other features. Next we can perform the following probabilistic queries:

Conditional probability queries: most linguistic universals are implicational and stated in conditional statements. In a PGM framework the question is formulated as conditional probability P(Q|C) where C is the condition variable and Q the query variable. We calculated all pairwise conditional probabilities using the UNIV model. The top ten pairs are listed in Table 1. We can also set multiple conditions at the same time. For example, when a language has values postposition and adjective preceding noun, in SamIam we set POST1 and AN1 as observed values and the probabilities of all other values are calculated as shown in Figure 5.

MAP queries aim to find the most likely joint assignment to a specific set of variables simultaneously given evidence. For example, given the value of O_V we can calculate the joint probability of having all other 12 values together.

Probability of evidence: If we calculate the joint probability of all 13 features then we have a probability indicating the likelihood of having all these feature values in a language, in other words, the likelihood of existence of such a language in terms of these features.

Prediction of unknown feature values of new languages: Here we use Hakka as an example. In the WALS database it has five feature values missing. The predicted values are listed in Table 2. According to the resources we found and also through elicitation from Hakka speakers, we filled in the missing values in the "TRUE" row. Both COMP and UNIV get 80% of the values correct while local model AREA6 get 60%. We have not done evaluation using datasets as described in (Bjerva et al., 2020) but plan to do it soon.

Prediction of macroarea: Finally given the values of a new language, we can also use our models to tell which macroarea it belongs to. Take Mandarin as an example: We calculate the probabilities of evidence using the six local models. The results are shown in Table 6. We can see that the local model for the area "Southeast Asia and Oceania" gives the highest probability of evidence therefore it is most likely that Mandarin belongs to this area.

References

Bjerva, J., Salesky, E., Mielke, S. J., Chaudhary, A., Celano, G. G. A., Ponti, E. M., Vylomova, E., Cotterell, R., & Augenstein, I. (2020). SIGTYP 2020 Shared Task: Prediction of Typological Features. *arXiv preprint arXiv:2010.08246.*

Croft, W. (2002). *Typology and universals.* Cambridge University Press.

Dryer, M. S. (1989). Large Linguistic Areas and Language Sampling. *Studies in Language. International Journal sponsored by the Foundation "Foundations of Language", 13*(2), 257-292.

Dryer, M. S., & Haspelmath, M. (Eds.). (2013). *WALS Online*. Max Planck Institute for Evolutionary Anthropology. https://wals.info/

Friedman, N. (1998). The Bayesian Structural EM Algorithm. UAI'98 Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, 129–138.

Friedman, N., Goldszmidt, M., & Wyner, A. (1999). Data Analysis with Bayesian Networks: A Bootstrap Approach. http://arxiv.org/abs/1301.6695

Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Appendices

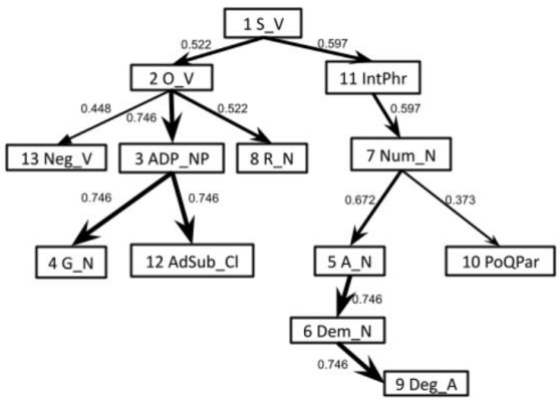

Figure 2: Structure of the UNIV model.

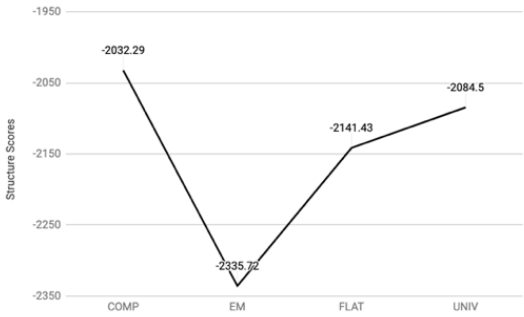

Figure 3: Structure scores of the four models.

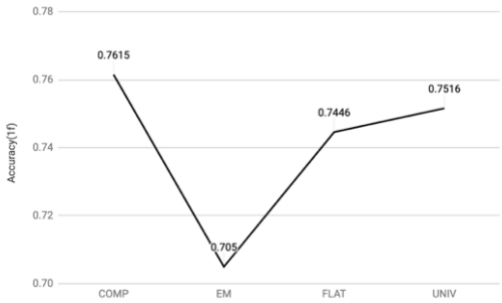

Figure 4: One-feature accuracy of the four models.

Query	Probability
p(O_V(OV1) -> S_V(SV1))	0.973
p(R_N(RN2) -> O_V(OV1))	0.964
p(R_N(RN2) -> S_V(SV1))	0.962
p(AdSub_Cl(Final2) -> ADP_NP(POST1))	0.957
p(R_N(CORRELATIVE4) -> S_V(SV1))	0.947
p(O_V(VO2) -> R_N(NR1))	0.947
p(R_N(IN_HEAD3) -> S_V(SV1))	0.945
p(S_V(VS2) -> O_V(VO2))	0.937
p(ADP_NP(PRE2) -> O_V(VO2))	0.930
p(R_N(CORRELATIVE4) -> O_V(OV1))	0.929

Table 1: The top 10 pairwise probability queries.

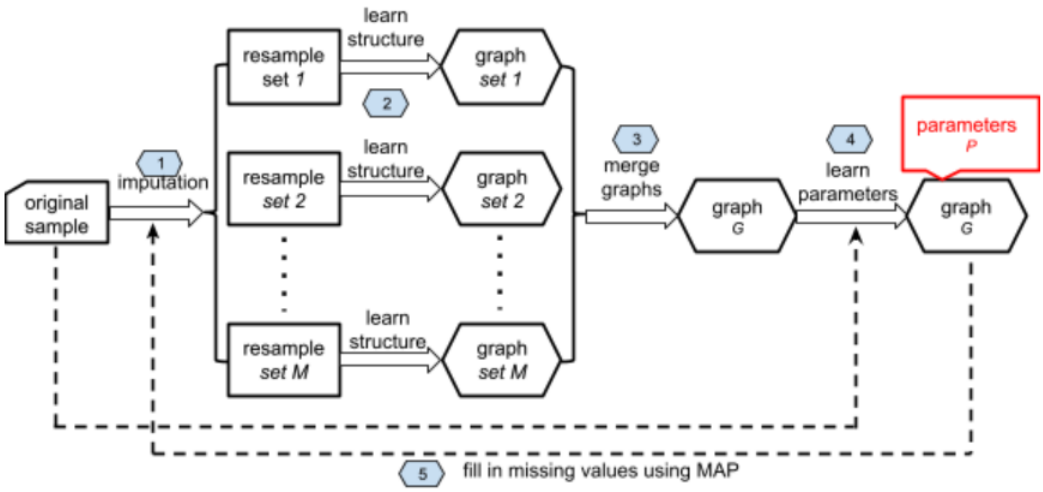

Figure 1: Illustration of the algorithm for learning the FLAT model.

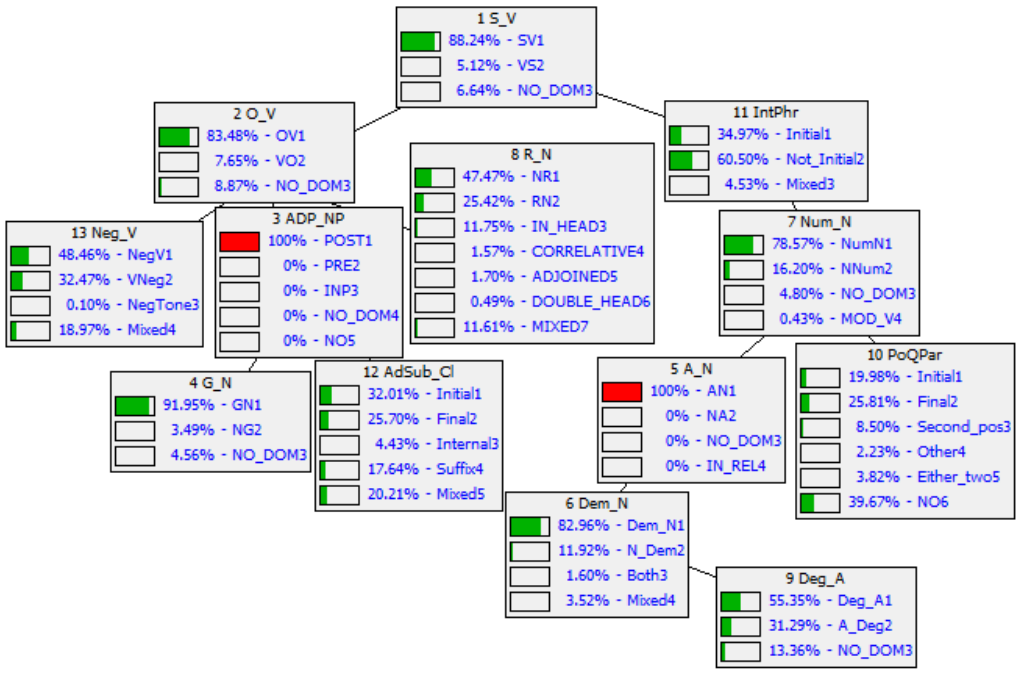

Figure 5: Example of probability query with multiple condition variables.

Hakka	S_V	O_V	ADP_NP	G_N	A_N	Dem_N	Num_N	R_N	Deg_A	PoQPar	IntPhr	AdSub_Cl	Neg_V
WALS	1	2				1	1	2	3		2		1
TRUE	1	2	1	1	1	1	1	2	3	2	2	1	1
COMP	1	2	1	1	1	1	1	2	3	1	2	1	1
AREA6	1	2	5	1	1	1	1	2	3	6	2	1	1
UNIV	1	2	1	1	1	1	1	2	3	1	2	1	1

Table 2: Prediction of missing values for Hakka.