Text-Based Joint Prediction of Numeric and Categorical Attributes of Entities in Knowledge Bases

V Thejas
BITS Pilani, India

Abhijeet Gupta and **Sebastian Padó**
IMS, University of Stuttgart, Germany

Abstract

Collaboratively constructed knowledge bases play an important role in information systems, but are essentially always incomplete. Thus, a large number of models has been developed for Knowledge Base Completion, the task of predicting new attributes of entities given partial descriptions of these entities. Virtually all of these models either concentrate on numeric attributes (`<Italy,GDP,2T$>`) or they concentrate on categorical attributes (`<Tim Cook,chairman,Apple>`).

In this paper, we propose a simple feed-forward neural architecture to jointly predict numeric and categorical attributes based on embeddings learned from textual occurrences of the entities in question. Following insights from multi-task learning, our hypothesis is that due to the correlations among attributes of different kinds, joint prediction improves over separate prediction.

Our experiments on seven FreeBase domains show that this hypothesis is true of the two attribute types: we find substantial improvements for numeric attributes in the joint model, while performance remains largely unchanged for categorical attributes. Our analysis indicates that this is the case because categorical attributes, many of which describe membership in various classes, provide useful 'background knowledge' for numeric prediction, while this is true to a lesser degree in the inverse direction.

1 Introduction

Collaboratively constructed knowledge bases (CCKBs) such as WikiData (Vrandečić and Krötzsch, 2014), YAGO (Suchanek et al., 2008), FreeBase (Bollacker et al., 2008) or DBPedia (Bizer et al., 2009), capture world knowledge in the shape of a graph structure where nodes denote *entities* and edges denote *attributes* (Hitzler et al., 2009). Their collaborative construction importantly enables them to avoid the scaling problems encountered by expert-constructed knowledge bases. Thus, CCKBs have come to play an important role in information systems, forming the basis for a wide range of natural language processing applications (Hovy et al., 2013) such as question answering (Berant et al., 2013; Krishnamurthy and Mitchell, 2015) or representation learning for entities (Toutanova et al., 2015; Yaghoobzadeh et al., 2018).

The most crucial shortcoming of CCKBs is their incompleteness (Min et al., 2013; West et al., 2014) – not just with respect to the entities that they cover, but also with respect to the attributes present for entities that are nominally covered. This is not surprising: When a contributor to a knowledge base adds an entity, they will probably concentrate on the most salient attributes (e.g., for a scientist, *field* or *affiliation*), while other attributes (such as *parents* or *place of birth*) may be added later or never. This realization has led to a large boost to work in the area of *knowledge base completion*, that is, the prediction of attributes of entities that are currently missing from the CCKB (Bordes et al., 2013; Socher et al., 2013; Min et al., 2013; Guu et al., 2015; Gupta et al., 2017).

These methods, however, overwhelmingly concentrate on *categorical* attributes, that is, attributes whose values are themselves entities in the knowledge graph. As an example, consider the attribute *capital* that maps a *country* onto a *city* which is itself an entity (*Mexico – Mexico City, UK – London*). A prominent approach to the prediction of categorical attributes is as an operation in embedding space, which explains the popularity of embedding-based approaches for this task.

Much fewer studies has considered the prediction of *numerical* attributes of entities in CCKBs (Davidov and Rappoport, 2010; Gupta et al., 2015),

Proceedings of Recent Advances in Natural Language Processing, pages 1194–1202,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_137

Attribute	Value
latitude	41.90 N
longitude	12.49 E
GDP_per_capita::2015	29,957.8 US$
fertility_rate::2010	1.46
capital	Rome
containedBy	Western_Europe
containedBy	Europe
member_of	G8
member_of	European_Union

Table 1: Sample of numeric and categorical FreeBase attributes for *Italy*.

such as the attribute *GDP-1990* which maps a *country* onto a number denoting its gross domestic product in the year 1990. For many entities in CCKBs, numeric attributes actually form the majority of the attributes for entities. Still, these attributes are often seen as secondary because their values are not 'proper' entities but numeric constants that themselves do not possess interesting attributes.

In this paper, we investigate the hypothesis that *joint prediction of numeric and categorical attributes* can improve prediction quality for both attribute types. As a motivating example, consider the sample of both numeric and categorical attributes listed for the country *Italy* in the FreeBase CCKB (Bollacker et al., 2008), shown in Table 1. It is clear that, as assumed by most models concentrating on categorical attributes, these attributes correlate with one another, and therefore the presence of one attribute can serve as evidence for the presence of another attribute. For example, containment in Western Europe implies containment in Europe, and is correlated with membership in the European Union. However, similar correlations arguably hold between categorical and numeric attributes. For example, the high GDP per capita constitutes evidence for Italy's membership in the G8 political forum, or vice versa, membership in the European Union and the G8 points towards a high GDP per capita. Similarly, Italy's latitude and longitude (defined by FreeBase to be the capital's geolocation) determine Rome as the country's capital, and vice versa.

Concretely, in this paper we adopt two previous embedding-based models for the individual prediction of numeric and categorical attributes from textual data, respectively. We define a novel simple joint model that predicts both attribute types concurrently (Section 2) and evaluate these models on a sample of seven FreeBase domains (Section 3) and find that prediction improves substantially for numeric attributes, but remains constant for categorical attributes (Section 4). Our analysis indicates that this is the case because numeric attributes that are difficult to predict from text-based embeddings are still often correlated with categorical attributes, and that can thus profit from joint training, while this is not true for categorical attributes.

2 Predicting Numeric and Categorical Attributes from Text

The majority of methods to predict attributes for entities in CCKBs are based on techniques from representation learning. Specifically, they us distributed representations (i.e., vectors, also called *embeddings*) to represent the entities, and sometimes also the attributes. Embeddings can be built from different sources, such as the knowledge bases themselves (Bordes et al., 2013; Guu et al., 2015; Lin et al., 2015), from text corpora that mention these entities (Socher et al., 2013; Krishnamurthy and Mitchell, 2015), or from both (Toutanova et al., 2015; Yaghoobzadeh et al., 2018).

In this paper, we use two prediction models that build on embeddings that were built from text corpora, following the widely successful assumption that text corpora implicitly contain a large amount of world knowledge that can be extracted by observing the contexts in which words are used (the so-called distributional hypothesis) (Firth, 1957; Miller and Charles, 1991; Turney and Pantel, 2010; Mikolov et al., 2013). The formulation of attribute prediction on top of precomputed embeddings enables us to use rather simple supervised neural model which are generally considered the state of the art for computational models in natural language processing.

2.1 Numeric Prediction

The first model is a feed-forward neural network, shown on the left-hand side of Figure 1. It builds on a study that used a logistic regression model to predict the values of numeric values, scaled to the interval (0;1) (Gupta et al., 2015) to avoid the excessive influence of outliers that linear regression is sensitive to. The model uses an n-dimensional entity embedding as its input which is mapped through a $\tanh$ nonlinearity onto an h-dimensional hidden layer, which in turn maps onto an $|N|$-dimensional

output layer (where $|N|$ is the number of the numeric attributes) using a sigmoid nonlinearity. In other words, each unit in the output layer corresponds to one numeric attribute, and the model predicts all numeric attributes simultaneously.

We use a variant of the mean cross-entropy loss function commonly used for logistic regression. Let $a \in A$ denote an attribute, $E(a) = \text{Tr}(a) \cup \text{Val}(a) \cup \text{Ts}(a)$ the set of entities for this attribute, partitioned into training, validation, and test sets, and $v_a(e)$ and $\hat{v}_a(e)$ the gold and predicted values for entity e, respectively. Then

$$L_{\text{num}} = -\sum_{a \in A} \frac{1}{|A||Tr(a)|} \sum_{e \in Tr(a)} \big(v_a(e) \log \hat{v}_a(e) + (1 - v_a(e)) \log(1 - \hat{v}_a(e))\big) \quad (1)$$

Even though simple, this model shows good performance in predicting numeric attributes of entities in CCKBs, since distributed representations implicitly capture a large amount of world knowledge (Gupta et al., 2015) and the hidden layer enables the model to exploit correlations among numeric attributes.

2.2 Categorical Prediction

The second model (Gupta et al., 2017) is another feed-forward neural network, shown on the right-hand side of Figure 1. Again, it uses a precomputed n-dimensional entity embedding as its input. Since this model predicts only the value of one categorical attribute at one time, this embedding is complemented by a representation of the attribute, realized as a one-hot vector whose dimensionality is the number of categorical attributes $|C|$.[1] Again, the input is first mapped onto an h-dimensional hidden layer and then onto an output layer, passing through a $tanh$ nonlinearity in both steps.

In this model, the output layer is n-dimensional, like the input, and actually represents an embedding of the attribute value. For example, given the embedding for *Italy* and the attribute *capital* as input, the model should predict the embedding for *Rome*. To map the output of the model back onto an explicit entity, we perform a nearest-neighbor retrieval in the space of precomputed embeddings, which is feasible with specialized indexes (Babenko and Lempitsky, 2016).

The loss function we use for this model is a contrastive variant of mean squared error (MSE) loss: we minimize the MSE between the prediction and the correct embedding while maximizing the MSE between the prediction and a sample of confounders. Since MSE can be understood as (squared) Euclidean distance, this loss function pushes the predicted embedding towards the correct embedding and away from confounders:

$$L_{\text{cat}} = \sum_{a \in A} \frac{1}{|A||Tr(a)|} \sum_{e \in Tr(a)} \big((v_a(e) - \hat{v}_a(e))^2 - \mu \sum_{e' \in \text{NN}(k, \hat{v}_a(e), Y - \{e\})} (v_a(e') - \hat{v}_a(e))^2\big) \quad (2)$$

The notation is the same as in Equation (1). Additionally, $\text{NN}(k, x, X)$ is a function that returns the k nearest neighbors of x in the set X, and μ a weight that trades off the positive and negative parts of the loss against each other. In this model, we do not need an indicator function as in the numeric attribute model, since the loss in this model is defined over seen attributes.

2.3 Joint Prediction

The similar structure of the two models described directly above makes it easy to define a joint model for the prediction of categorical and numeric attributes, shown in Figure 2. The new architecture re-uses the input layer from the categorical model, which subsumes the simpler architecture of the numeric one. It uses the same type of hidden layer, to which both the numeric output layer and the categorical output layer are attached. The nonlinearities are the same as in the individual models. Since the input to the model is still an entity embedding plus a categorical attribute, as in the categorical model, the model essentially predicts the numeric attributes of the entity "on the side".

Correspondingly, the loss function of this model is a weighted average of the losses of its parts:

$$L_{\text{joint}} = \alpha L_{\text{cat}} + (1 - \alpha) L_{\text{num}} \quad (3)$$

where α is the relative weight of the categorical loss. For the extreme values of $\alpha = 1$ and $\alpha = 0$, the joint model reverts to its component models.

[1] We experimented with learning a distributed representation of the attributes, but did not achieve better results.

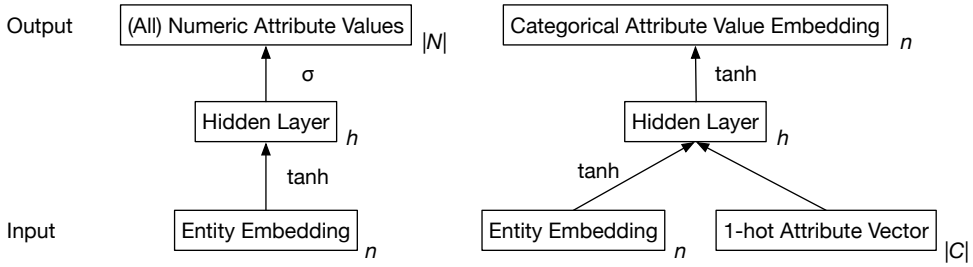

Figure 1: Individual model architectures. Prediction of numeric attributes (left-hand side) and categorical attributes (right-hand side). Subscripts in italics indicate dimensionality of layers (n: dimensionality of embedding space; h: dimensionality of hidden layer; $|N|$: number of numeric attributes, $|C|$: number of categorical attributes)

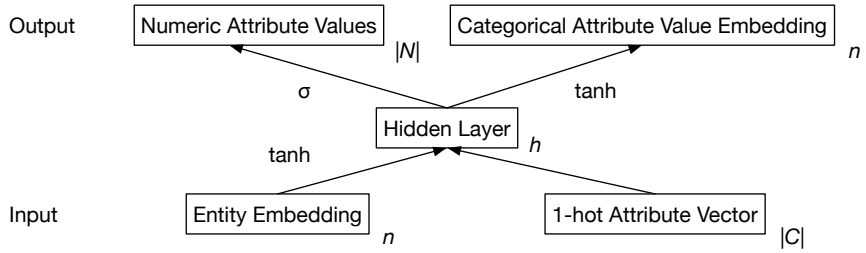

Figure 2: Joint model architecture for the simultaneous prediction of categorical and numeric attributes. Subscripts in italics indicate dimensionality of layers (n: dimensionality of embedding space; h: dimensionality of hidden layer; $|N|$: number of numeric attributes, $|C|$: number of categorical attributes)

Our hypothesis is that intermediate values of α will improve prediction quality for the two types of attributes. We expect this to be the case since joint training can be seen as an instance of multi-task learning, which is known to often positively impact the quality of the learned intermediate representations (Zhang and Yang, 2017). Note that this effect is not guaranteed, since we introduce competition among the two output layers, which may deteriorate the output of the 'losing' layer.

2.4 Discussion

Note that all three model assume that all entities share a common set of attributes: in the numeric model, these determine the shape of the output layer, and in the categorical mode, they determine the shape of the attribute input layer. While it is still possible to train global models, CCKBs are typically organized into top-level *domains* that share little to no attributes. For example, *people* (which have, e.g., birth and death dates) or *organizations* (which have e.g., personnel, turnover, profit numbers) have no attributes in common. Consequently, in the remainder of the paper, we adopt a *domain*-specific approach, learning and evaluating separate models for each domain.

3 Experimental Setup

3.1 Dataset and Embeddings

To our knowledge, there are no existing datasets that include both numeric and categorical attributes. For example, the widely used FB15K and WN18 datasets (Bordes et al., 2013) focus exclusively on categorical attributes. For this reason, we construct our own dataset which we make freely available on DANS at URL `https://doi.org/10.17026/dans-zxp-t7tf`.

We construct the dataset on the basis of the FreeBase CCKB (Bollacker et al., 2008). As sketched above, we proceed by domains and extract entities and attributes for six of the most populous top-level FreeBase domains (*animal, book, citytown, country, employer, organization, people*).

Since we build on pretrained embeddings for the entities in question, we only include entities if they are covered by the largest existing pretrained embedding space for proper names. This is the "Google News" embedding space that used a 100G token news corpus to compute embeddings specifically for FreeBase entities (Mikolov et al., 2013).[2] The embeddings are computed with the Word2Vec

[2] `https://code.google.com/p/word2vec/`

skip-gram algorithm, 1000 dimensions. Similarity, categorical attributes are only included if we have embeddings for both the entity and the value.

Finally, we split all domains into training, validation, and test sets (60%–20%–20%). The split is applied to each attribute type: at validation and test time, our models face no unseen attribute types, but unseen instances for each attribute. In the numeric and joint models, this means that the model will encounter 'incomplete' numeric output layers since some attributes of a given entity may be reserved for testing (cf. the left-hand side of Figure 1 as well as Figure 2). This does not hurt the model, though: The objective function, Equation (1), only ranges over attributes present in the training data.

Table 2 shows descriptive statistics for the resulting dataset. We consider just over 5000 entities for a total of 269 categorical attribute types and 1041 numeric attribute types.[3] Note that the domains differ considerably with regard to their numbers of entities, numbers of attributes, and relative prevalence of categorical and numeric attributes. For example, the *country* domain has the highest number of attributes, and about ten times as many numeric as categorical attributes. This reflects the large number of time series recorded for countries. In contrast, the *organization* domain has much fewer attributes overall, and more categorical than numeric attributes (e.g., location, founders, officers, business sector).

3.2 Evaluation

Categorical Attributes. As explained above in Section 2.2, we apply nearest neighbor mapping to the embedding output of the model to map its prediction back onto an entity symbol. Following earlier work (Gupta et al., 2017), we perform Information Retrieval-style ranking evaluation, mean reciprocal rank (MRR) (Manning et al., 2008). Reusing the notation from Sec. 2 and writing ra for rank, we define MRR as

$$\frac{1}{|T|}\sum_{a\in A}\sum_{e\in \mathrm{Ts}(a)}\frac{1}{\mathrm{ra}(\hat{v}_a(e), \mathrm{NN}(\infty, v_a(e), v_a(\mathrm{Ts}(a))))}$$

For each entity-attribute pair (e, a), MRR computes the (reciprocal) rank of the model's prediction $\hat{v}_a(e)$ in the nearest neighbor list of the true value $v_a(e)$. These values are averaged over all datapoints in the test set Ts.

[3]We removed attributes that were not populated for any entities in our entity set.

Intuitively, MRR describes how close, on average, the predictions are to the correct one in terms of ranks: an MRR of 0.5 means that they are the second-nearest neighbors, an MRR of 0.3 means that they are the third-nearest neighbors, and so on. Thus, higher MRR values indicate better performance. We report results at the domain level as well as micro-averaged MRR for the complete dataset.

Numeric Attributes. For numeric attributes, we use the so-called normalized rank score (NRS). NRS is a variant of Spearman's correlation coefficient that takes into account both how correctly the entities in the test set are ranked with respect to each numeric attribute, and how consistent the predictions are with regard to the training set (Frome et al., 2013). We choose this evaluation over a numeric error-based one because it is more robust to outliers and sets a more realistic target for the prediction of numeric attributes (Gupta et al., 2015). NRS is defined as

$$\sum_{a\in A}\frac{1}{|A||\mathrm{Ts}(a)|}\operatorname*{med}_{e\in\mathrm{Ts}(a)}\{|\mathrm{ra}(\hat{v}_a(e), E(a)) - \mathrm{ra}(v_a(e), E(a))|\}$$

NRS measures divergence from the gold standard ranking. It has range [0;1], with smaller numbers indicating better performance: 0.2, for example, means that the prediction is, on average, off by 20% of the ranks. As before, we report the statistic for each domain, plus a micro-averaged NRS for the complete dataset.

3.3 Hyperparameters

Individual Models. We trained the two individual models and the joint model using AdaDelta optimization method, using the best parameters according to the literature (Zeiler, 2012), namely $\rho = 0.95$ and $\epsilon = 10^{-6}$. We trained until convergence or for at most 300 iterations with early stopping. All hyperparameters were explored on the validation set. We explored h, the size of the hidden layer, by setting it to values between 200 and 3000 with a step size of 200. We found h=2000 to yield good results for both models and adopted this number. In the model for categorical attributes, we followed earlier work (Gupta et al., 2017) by using just a single nearest neighbor for the negative part of the loss (k=1) and setting μ to 0.6.

Joint Model. To build the joint model, we retained the hyperparameter settings of the two individual models. We explored values of α between

Domain	# Entities (train/val/test)	$\|C\|$	$\|N\|$
Animal	279/93/93	22	118
Book	16/5/6	8	2
Citytown	1783/594/595	57	62
Country	155/53/51	79	698
Employer	720/140/141	50	55
Organization	187/63/62	36	32
People	85/28/29	25	76
Sum	3225/976/977	277	1043

Table 2: Data set statistics. $|C|$: number of categorical attribute types. $|N|$: number of numeric attribute types

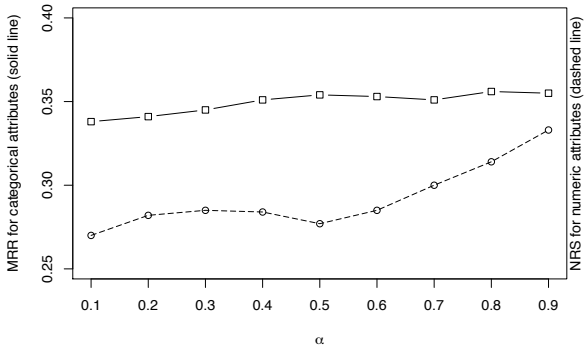

Figure 3: Hyperparameter exploration. Impact of different values of α on the *animal* domain for categorical (solid) and numeric (dashed) attributes (validation set).

0.1 and 0.99 on the validation set of the *animal* domain. The results for the joint model are shown in Figure 3. As expected, there is a trade-off between the two objectives: Results for categorical prediction improve for high values of α, where the model focuses on these attributes. Conversely, results for numeric prediction improve when the model pays more attention to these attributes, for low values of α (recall that lower NRS values are better). We chose $\alpha = 0.6$ as a value that gives both models a chance to profit from the joint setup.

3.4 Inference

Regarding inference in the models, the two individual prediction models are trivial, and so is the categorical part of the joint model:To predict the value of a categorical attribute for an entity, the numeric output can simply be ignored. To predict the value of a numeric attribute of entity, however, different inference procedures are possible. We used the simplest one, namely activating a random categorical attributes to query a numeric attribute (cf. Figure 2). We did not observe meaningful variance across the choice of different categorical attributes.

3.5 Baselines

We use two baseline models from previous studies. For categorical attributes, our baseline model ignores the entity. For each attribute, it predicts the frequency-ordered list of all values seen in the training set (Frequency Baseline). We also report on a baseline that simply models each attribute as a linear operation in embedding space (Mikolov et al., 2013; Bordes et al., 2013) defined as the centroid of all difference vectors for a given attribute between entities and their values for this attribute (Linear Baseline).

For numeric attributes, our baseline model predicts the mean value of the attribute seen in the training set (Mean Baseline).

4 Results and Discussion

Numeric Attributes. Table 3 shows the results as averaged normalized rank scores (NRS) for each domain as well as macro-averaged (Avg) scores for the complete test set. Recall that for NRS lower values are better.

We find that that the joint model (which predicts numeric and categorical attributes at the same time) yields substantially better results than the individual model on all domains, ranging between 0.03 (for *animal* and *people*) and 0.1 (*books*). Average performance on all domains improves from 0.3 by 0.07 to 0.23. In turn, the individual model outperforms the baseline on all domains except *people*, corresponding to a similar improvement by 0.07.

We see the best results of the joint model for *citytown* and the worst results for *people*. These numbers correlate with the numbers of entities present

	Domains							Mean
	Animal	Book	City	Country	Employer	Organization	People	
Joint Model	**0.284**	**0.276**	**0.211**	**0.293**	**0.215**	**0.225**	**0.387**	**0.229**
Individual Models	0.317	0.382	0.288	0.376	0.289	0.300	0.421	0.300
Mean BL	0.370	0.434	0.366	0.416	0.364	0.421	0.394	0.373

Table 3: Test set results on numeric attributes per domain (normalized rank score; lower is better). Best result for each domain marked in boldface.

	Domains							Mean
	Animal	Book	City	Country	Employer	Organization	People	
Joint Model	0.330	0.244	0.198	**0.105**	**0.118**	**0.096**	**0.352**	0.193
Individual Models	**0.331**	**0.256**	**0.202**	**0.105**	0.116	**0.096**	**0.352**	**0.195**
Linear BL	0.215	0.217	0.084	0.046	0.085	0.090	0.250	0.101
Frequency BL	0.247	0.225	0.045	0.018	0.028	0.022	0.173	0.064

Table 4: Test set results on categorical attributes per domain (mean reciprocal rank; higher is better). Best result for each domain marked in boldface.

for these domains (Table 2): citytown is the largest domain, with almost 1800 entities in the training set, while people is the smallest domain with 85 training entities. Thus, we surmise that the differences in performance are not due to inherent differences in the features to be predicted, but reflect the different amounts of training data available.

Categorical Attributes. Table 4 shows the results as mean reciprocal rank (MRR) scores for each domain as well as macro-averaged scores (Avg) for the complete test set. For MRR, higher values are better.

The joint and the individual model outperform both baselines for all domains, and outdistance them substantially in average performance. However, the two informed models show almost identical performance overall, with average MRRs of 0.193 and 0.195, respectively. For three domains (*country, organization, people*), they perform equally well. For one domain (*employer*), the joint model performs better, and for three domains (*animal, book, city*), the individual model does better. The results of the two models are generally extremely close to one another, with the largest difference between the models (of 0.012) appearing for *book*, the smallest domain where we would expect the largest variance. Overall, we attribute these differences to random fluctuation.

The performance of all models is markedly different across domains, with low results close to 0.1 for *organization* and *country* and high results over 0.3 for *animal* and *people*. This is in line with earlier results for categorical attributes that identified the predominance of one-to-many relations across domains as important predictor of performance (Gupta et al., 2017).

Discussion. Probably the most surprising outcome of our experiment is the asymmetry that we observe: Joint modeling achieves respective improvements over individual modeling for numeric attributes, but not for categorical attributes. In other words, the prediction of numeric attributes profits from the availability of categorical attributes, but not vice versa.

One potential explanation is a suboptimal setting of the α parameter (Equation (3)) that would let the joint model pay too much attention to the numeric attributes. However, Figure 3 indicates that this not the case: the change in performance on categorical attributes is relatively minor across values of α, in particular compared to the chance in performance on numeric attributes.

To search for alternative explanations, we performed a qualitative analysis of the models' predictions. What we observe is that many numeric attributes of entities have a relatively low degree of *contextual support* (Gupta et al., 2015), that is, the values of these attributes do not correlate well with salient textual characteristics of the occurrences of the entity name. For example, in the *animal* domain, attributes like 'life span' or 'litter size' describe relatively detailed properties of

animal species. Such attributes are unlikely to be represented well in embeddings learned in an unsupervised manner from newspaper text. As another example, the *country* domain contains attributes like 'diesel price' or 'gender balance among members of parliament' that arguably suffer from the same problem.

We believe that the prediction of such attributes can profit from information added to the hidden layer by the categorical part of the objective (Fig. 2), since specific values of categorical attributes provide informative priors. For animals, life span and litter size differ, for example, among different animal classes, orders, etc.; for countries, fuel prices or gender equality are correlated with categorical attributes such as membership in organisations (OPEC, Nordic Council).

For categorical attributes, an earlier study (Gupta et al., 2017) found that difficulty arose both from lack of contextual support and from list-valued attributes. In contrast to the numeric side, though, lack of support for categorical attributes can often not be compensated by access to numeric information. An examples, consider attributes such as 'disputed territories' from the *country* domain, or 'supplier' from the *organization* domain; arguably the values of these attributes is so specific that numeric information cannot help. Nor can the fundamental problem of list-valued attributes be alleviated by numeric information. Instead, this would require a fundamentally different prediction mechanism that supports list-valued attributes (Lin et al., 2015).

5 Conclusion

This paper is located in the area of knowledge base completion, that is, the task of complementing knowledge bases with missing relations, which is particularly pressing for collaboratively constructed knowledge bases. We focus on an understudied subproblem of knowledge base completion, namely the prediction of numeric, as opposed to categorical, attributes.

We assume a text-based approach that uses corpus-derived entity embeddings as the basis for attribute prediction. Building on top of two existing models for categorical and numeric attributes, the first contribution of this paper is a joint model for the prediction of these two attribute types. The second contribution is an empirical evaluation of separate vs. joint modeling on a novel dataset, where we find that numeric attributes profit substantially from a joint model, while categorical attributes do not. A qualitative analysis of the predictions indicates that there is indeed an asymmetry: in cases where the values of numeric attributes are difficult to predict from text-based embeddings, categorical information about the entity can often serve as a prior, whereas difficult-to-predict categorical attributes are often so specific that numeric attributes do not help.

To the best of our knowledge, this paper presents the first joint model for the prediction of numeric and categorical attributes. The joint model that we present is a straightforward combination of individual models for the two attribute types, both of which are purely text-based. A first step to improve the models would be to learn, or at least fine-tune, text-based embeddings in a task-specific manner, in order to enable the embeddings to pay attention to infrequent context cues that are nevertheless highly informative for particular attributes. On a more fundamental level, embeddings can be made to take both textual evidence and the structure of the knowledge base into account, as has been demonstrated for categorical attributes (Toutanova et al., 2015; Yaghoobzadeh et al., 2018). Finally, a direction for future research that would address in particular the difficulties in predicting categorical attribute could be the development of a neural architecture that explicitly accounts for list-valued attributes.

References

Artem Babenko and Victor S. Lempitsky. 2016. Efficient indexing of billion-scale datasets of deep descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2055–2063, Las Vegas, NV.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of EMNLP*, pages 1533–1544, Seattle, WA.

Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. 2009. DBpedia – A crystallization point for the Web of Data. *Journal of Web Semantics*, 7(3):154–165.

K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of ACM SIGMOD*, pages 1247–1249, Vancouver, Canada.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of NIPS*, pages 2787–2795, Lake Tahoe, NV.

Dmitry Davidov and Ari Rappoport. 2010. Extraction and approximation of numerical attributes from the web. In *Proceedings of ACL*, pages 1308–1317, Uppsala, Sweden.

J. R. Firth. 1957. A synopsis of linguistic theory 1930–55. In *Studies in Linguistic Analysis*, pages 1–32. The Philological Society, Oxford.

Andrea Frome, Greg Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. 2013. DeViSE: A deep visual-semantic embedding model. In *Proceedings of NIPS*, pages 2121–2129, Lake Tahoe, NV.

Abhijeet Gupta, Gemma Boleda, Marco Baroni, and Sebastian Padó. 2015. Distributional vectors encode referential attributes. In *Proceedings of EMNLP*, pages 12–21, Lisbon, Portugal.

Abhijeet Gupta, Gemma Boleda, and Sebastian Padó. 2017. Distributed prediction of relations for entities: The easy, the difficult, and the impossible. In *Proceedings of STARSEM*, pages 104–109, Vancouver, BC.

Kelvin Guu, John Miller, and Percy Liang. 2015. Traversing knowledge graphs in vector space. In *Proceedings of EMNLP*, pages 318–327, Lisbon, Portugal.

Pascal Hitzler, Markus Krotzsch, and Sebastian Rudolph. 2009. *Foundations of semantic web technologies*. CRC Press.

Eduard Hovy, Roberto Navigli, and Simone Paolo Ponzetto. 2013. Collaboratively built semi-structured content and artificial intelligence: The story so far. *Artificial Intelligence*, 194:2–27.

Jayant Krishnamurthy and Tom M Mitchell. 2015. Learning a compositional semantics for freebase with an open predicate vocabulary. *Transactions of the Association for Computational Linguistics*, 3:257–270.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of AAAI*, Austin, TX.

Chris Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111–3119, Lake Tahoe, NV.

G Miller and W Charles. 1991. Contextual Correlates of Semantic Similarity. *Language and Cognitive Processes*, 6(1):1–28.

Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. 2013. Distant supervision for relation extraction with an incomplete knowledge base. In *Proceedings of NAACL-HLT*, pages 777–782, Atlanta, Georgia.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Proceedings of NIPS*, pages 926–934, Lake Tahoe, CA.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. YAGO: A Large Ontology from Wikipedia and WordNet. *Journal of Web Semantics*, 6:203–217.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Proceedings of EMNLP*, pages 1499–1509, Lisbon, Portugal.

Peter D Turney and Patrick Pantel. 2010. From Frequency to Meaning: Vector Space Models of Semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledge base. *Communications of ACM*, 57(10):78–85.

Robert West, Evgeniy Gabrilovich, Kevin Murphy, Shaohua Sun, Rahul Gupta, and Dekang Lin. 2014. Knowledge base completion via search-based question answering. In *Proceedings of WWW*, pages 515–526, Seoul, Korea.

Yadollah Yaghoobzadeh, Heike Adel, and Hinrich Schütze. 2018. Corpus-level fine-grained entity typing. *Journal of Artificial Intelligence Research*, 61:835–862.

Matthew D. Zeiler. 2012. Adadelta: An adaptive learning rate method. In *CoRR, abs/1212.5701*.

Yu Zhang and Qiang Yang. 2017. An overview of multi-task learning. *National Science Review*, 5:30–43.