Information Extraction with Weak Supervision

Keywords information networks, knowledge base completion, slot-filling, natural language processing, information extraction, latent variable models

Relevance to Army BAA In order to provide decision makers with the information necessary to make informed choices as well as predict the effects of those choices, information networks must 1) contain correct information in each node, as well as 2) represent the causal relationships between nodes efficiently in order to propagate changes dynamically, where we define information networks to be graphical representations of data where each node contains a set of attributes and edges between nodes determine how information should propagate. We argue that 1) alone is difficult to satisfy at scale. Removing the capacity to propagate information between nodes in an information network yields a "knowledge base," which is a structured representation of data where tuples of entities, predicates, and values are stored. To populate knowledge bases with correct data, we must either train systems using explicit supervision or formulate models that are able to extract information from multiple and possibly unreliable sources. As acquiring explicit supervision becomes extremely expensive as the size of the knowledge base grows, we present a framework towards automating the training of information extraction systems for the purpose of populating knowledge bases with minimal supervision. In this proposal we focus on slot-filling, the task of filling in the missing or incorrect values of a knowledge-base, while leaving other tasks as possible extensions to our proposed model.

Introduction Since the addition of neural networks to the practitioner's toolbox, extracting information from text has been cast as a supervised classification problem. In fact, in many tasks where generative models were once used, such as coreference and slot-filling, recent approaches make progress by fitting large models to large, labelled datasets [4, 9]. Although this does result in progress, this particular strategy is not scalable as acquiring supervision is expensive and time-consuming. An orthogonal strategy is to search for a class of models that is able to learn well under less supervision. Latent variable models (LVMs) in particular lend themselves well to label-efficient semi-supervised learning. A LVM is a model which includes both observed and unobserved (latent) random variables. Past work on coreference and slot-filling utilized LVMs in order to train with distant supervision. More specifically, Haghighi and Klein [3] learn a semi-supervised generative model of mentions given mention segmentations and a relatively small set of labels, while Surdeanu et al. [6] learn to extract the relationships between two entities with noisy labels. We propose to unify past work on LVMs for information extraction with the powerful modeling capabilities of neural networks.

Recent work has demonstrated that LVMs parameterized with neural networks can be trained efficiently. In machine translation, we demonstrated in Deng et al. [1] that formulating attention as a latent variable provides a boost in performance without increasing the computational complexity of training. The specific model we used did not incorporate interactions between the latent variables, however the technique we used generalizes to classes of LVMs that do specify interactions. In this proposal, we focus on the class of LVMs known as hidden semi-Markov models (HSMMs), used in Liang et al. [5] as a generative model for the task of aligning segments of text to records in a knowledge base without supervision. As in Liang et al. [5], we are interested in learning a generative model of text so that we can minimize the amount of supervision necessary for training an information extraction system. The performance of their generative model of text provides signal for learning the alignments: if the likelihood of a segment of text improves when moving from a past alignment choice to a new one, then the new alignment is more likely to be correct given a suitably strong likelihood model. We use that same intuition to formulate a related LVM for weakly supervised information extraction which aims to model not just the alignments from segments of text to records in a knowledge base but also the values in the knowledge base itself. By parameterizing the generative HSMM with neural networks as in Wiseman [7], we hope to incorporate recent progress in parameterizing LVMs with neural networks so as to learn a more accurate information extraction system by using a more powerful generative model – taking care to ensure that the model can be trained with minimal supervision.

Background We consider datasets consisting of aligned data and text $\{(\mathbf{r}^{(1)}, \mathbf{y}^{(1)}), (\mathbf{r}^{(2)}, \mathbf{y}^{(2)}) \ldots\}$. For brevity, we refer to a single datum and text as $\mathbf{r}, \mathbf{y}$, omitting the superscript. Each datum $\mathbf{r} = \{r_1, \ldots, r_N\}$ is a set of N records, where each record $r_i = (e_i, t_i, v_i)$ is a tuple containing an entity, type, and value. The datum $\mathbf{r}$ is a knowledge base, or equivalently an information network without a representation of the causal effects between records. We refer collectively to all the entities, types, and values in a given datum $\mathbf{r}$ as $\mathbf{e}, \mathbf{t}, \mathbf{v}$ respectively. Each text $\mathbf{y} = \{y_1, \ldots, y_T\}$ is a sequence of tokens each from a vocbulary V.

Let variables $\mathbf{z}$ be unobserved or latent, $\mathbf{y}$ observed, and $\mathbf{x}$ taken as conditioning and thus not modelled. For information extraction we are interested in distributions that can be specified as $p(\mathbf{z} \mid \mathbf{y}, \mathbf{x})$, where $\mathbf{z}$ and $\mathbf{x}$ may correspond to various quantities depending on the task but $\mathbf{y}$ is always the text.

As a concrete example, we use the Rotowire dataset [8]. Rotowire contains summaries of basketball games $\mathbf{y}$ aligned with the respective box scores $\mathbf{r}$ of those games. Consider a datum that consists of a single record, $\mathbf{r} = \{(e_1 = \text{Jeremy\_Lin}, t_1 = \text{POINTS}, v_1 = 19)\}$, and a simple statement $\mathbf{y} =$"Jeremy Lin scored 19 points". In its simplest incarnation, the process of information extraction may be to infer any subset of $\mathbf{r}$, which in this case will be our latent $\mathbf{z}$, given the remaining elements in $\mathbf{r}$ which corresponds to $\mathbf{x}$, as well as the text $\mathbf{y}$. For example, we may want to infer the type of the relation t given the entity Jeremy Lin, the value 19, as well as the text $\mathbf{y}$. In this case, we would have $\mathbf{z} = \{t_1\}$ and $\mathbf{x} = \{e_1, v_1\}$. In an alternative task, we may want to infer the value v_1 as well as the type t_1 given $\mathbf{y}$ and e_1, therefore $\mathbf{z} = \{t_1, v_1\}$ and $\mathbf{x} = \{e_1\}$.

Note that we are not constrained to setting $\mathbf{z}$ to subsets of $\mathbf{r}$. We also consider the case where $\mathbf{z}$ includes alignments from individual words y_t to records r_i. We denote the alignments $\mathbf{a} = \{a_1, \ldots, a_T\}$, where each a_t is associated with y_t and selects a record r_i such that $a_t = i$. (NEED TO FIX ALIGNMENTS AND T vs t, HMM vs HSMM)

Proposal We propose to verify the efficacy of the LVM framework in the weakly supervised information extraction setting, with the goal of demonstrating strong extractive performance with minimal labels. We present one instance of a LVM and outline how it can be used to obtain an information extraction model without direct supervision, then argue that the same approach can be applied in even more ambitious settings. Our first LVM is a conditional model that specifies the relationship between data, specifically the entities and types, and text. We denote this model `Values`. Similar to the models defined in Wiseman [7] and Liang et al. [5], our model takes the form of a hidden semi-Markov model (HSMM). The primary difference is that the other models simply assumed the records were complete and conditioned on them, whereas ours learns to generate the values. `Values` is given by the following generative process:

1. Value Choice: For each pair of entities and types in our datum of records, we predict a value. We assume that each record type constrains the values to be members of a finite set. Thus each record type is assigned a categorical distribution over its respective values, and the values

are drawn independently from that respective distribution. Each $v_i \sim \text{Cat}(f_{t_i}(e_i))$ is drawn from a Categorical distribution, whose parameters $f_{t_i}(e_i)$ are output by a neural network f_{t_i} that is shared across record types and takes as input the entity e_i.

2. Record Choice: Conditioned on our choices of values as well as the given entities and records, we choose a sequence of records $\mathbf{a} = \{a_1, \ldots, a_I\}$ to describe, given by their index a_i. Note that each record choice is described by at least one token, hence we have $I \leq T$. The record choices are parameterized as a Markov model where each $a_t \sim \text{Cat}(f_\theta(a_{t-1}, \mathbf{v}, \mathbf{e}, \mathbf{t}))$, where f_θ is a neural network. (NEED TO FIX ALIGNMENTS AND T vs t, HSMM vs HMM)

3. Word Choice: For each record alignment a_i, we choose a sequence of words $\mathbf{y}_i = \{y_{i1}, \ldots, y_{iJ}\}$ to describe the record. The words are modelled by an autoregressive emission model within each segment that is aligned to the same record: $y_{ij} \sim \text{Cat}(f_\theta(\mathbf{y}_{i1:ij}, v_{a_i}, e_{a_i}, t_{a_i}))$, where f_θ is another neural network and $\mathbf{y}_{i1:ij}$ is all tokens y from indices $i1$ to ij.

The value and record choices correspond to prior distributions over values $p(\mathbf{v} \mid \mathbf{e}, \mathbf{t})$ and alignments $p(\mathbf{a} \mid \mathbf{v}, \mathbf{e}, \mathbf{t})$ respectively, while the word choice model gives us the likelihood of some text given our value and alignment choices $p(\mathbf{y} \mid \mathbf{a}, \mathbf{v}, \mathbf{e}, \mathbf{t})$. In this case, we have latent $\mathbf{z} = \{\mathbf{v}, \mathbf{a}\}$ and observed $\mathbf{x} = \{\mathbf{e}, \mathbf{t}\}$. We obtain an information extraction by using the **posterior** distribution over alignments and values:

$$p(\mathbf{z} \mid \mathbf{y}, \mathbf{x}) = \frac{p(\mathbf{y}, \mathbf{z} \mid \mathbf{x})}{p(\mathbf{y} \mid \mathbf{x})} = \frac{p(\mathbf{y}, \mathbf{z} \mid \mathbf{x})}{\sum_{\mathbf{z}} p(\mathbf{y}, \mathbf{z} \mid \mathbf{x})}.$$

Although the HSMM formulation allows the summation (marginalization) over alignments to be carried out efficiently, we cannot marginalize over value assignments. We instead resort to variational inference as in Deng et al. [1], where we learn an approximation of the posterior distribution $q(\mathbf{z} \mid \mathbf{y}, \mathbf{x})$ with a separate model. Given that we are primarily interested in the values rather than the alignments, we can obtain an information extraction system over only values by marginalizing over alignments. By marginalizing over the alignment distribution, the model propagates uncertainty over alignments to uncertainty over values. We can train this model by maximizing a lower bound on the log marginal likelihood or evidence of $\mathbf{y}$, called the evidence lower bound (ELBO) which is given formally by the expression $\text{ELBO}_q \triangleq \mathbb{E}_{q(\mathbf{z}|\mathbf{y},\mathbf{z})}\left[\log p(\mathbf{y} \mid \mathbf{z}, \mathbf{x})\right] \leq \log \mathbb{E}_{p(\mathbf{z}|\mathbf{x})}\left[p(\mathbf{y} \mid \mathbf{z}, \mathbf{x})\right]$. This objective can be maximized with gradient-based methods. The resulting approximate posterior $q(\mathbf{z} \mid \mathbf{y}, \mathbf{x})$ can be used independently of the generative model as an information extraction system that gives a distribution over values in a table of records and alignments from text to records.

As we are interested in applying the LVM framework to information extraction rather than a single model, we proceed to motivate and outline an extension to `Values`. In addition to inferring values, ideally an information extraction system would also be able to infer new entities and relation types. We propose a possible route towards defining model that can learn to parameterize new relation types in an unsupervised manner using the same LVM framework as `Values`, which we denote `Types`. The motivation behind `Types` is that a segment of text may refer to multiple records at the same time. For example, in basketball games a 'triple-double' refers to more a player achieving a value of more than 10 in any three of the five categories: points, steals, rebounds, blocks, or assists. The goal of `Types` is to attempt to capture the latent relationship behind utterances such as 'triple-double' appearing in the text and the underlying records in an unsupervised manner. We plan to introduce a new step to the generative process of `Values` that allows the model to learn new records as boolean-valued functions of relations already defined in the data. By approximating these boolean functions with neural networks, we hope to find a model parameterization that admits a low variance Monte Carlo gradient estimator.

We will evaluate our initial approach on the Rotowire dataset, and extensions to our model that will include entity tracking and event resolution on the automatic content extraction (ACE) [2] and the Text Analysis Conference's Streaming Multimedia Knowledge Base Population (SM-KBP) datasets. We expect the variance of the gradient estimator to be an issue, in particular its effect on sample complexity. In previous work, we observed that gradient estimators based on exact inference resulted in better sample complexity than Monte Carlo gradient estimators [1], and we expect that controlling the variance of the gradient estimator with the inductive bias of neural architectures will be of paramount importance for the success of our proposed method.

Given admittance to the NDSEG Fellowship Program, I will evaluate the application of LVMs to the problem of information extraction. As a result of the digital age, the ubiquity of information networks as well as their enormous growth makes it clear that a method for training information extraction systems with minimal supervision is a necessity. I will push for scalable information extraction systems that require minimal supervision by recasting information extraction as a generative modeling problem.

Outline

1. Introduction

   (a) Relevance to BAA

      i. What are information networks? Graphical representation of objects, their characteristics of interest, and their relationship to objects.

      ii. Information networks are dynamic knowledge graphs? They must also capture dynamic relationships, i.e. the influence of changing an attribute of one node on another node.

      iii. Reduce information networks to knowledge bases by removing edges (kind of interesting as we can view knowledge bases as a mean field approximation to information networks where info networks must specify not only conditional distributions but also interventional distributions).

      iv. Obtaining supervision for large KBs and thus info networks is expensive, especially if the underlying schema is subject to change.

      v. In this proposal we present a framework towards automating the training of information extraction systems with minimal supervision.

      vi. Shine some hope on interventional distribution representations? (This might be a little difficult)

   (b) Supervision in Information Extraction

      i. Recent neural approaches treat information extraction as a classification problem with little intermediate structure.

      ii. Older approaches utilized latent variable models to capitalize on intermediate structure, however this was before neural models were used in NLP. [5, 6]

      iii. We aim to unify these two approaches through recently developed techniques for training LVMs parameterized with neural networks.

   (c) Recent advances in neural LVMs

      i. Demonstrate that parameterization with a neural network does not affect computational complexity of inference.

      ii. Then the same technique can be applied to model with more structure, as long as the graphical model itself permits tractable inference.

      iii. In this proposal, we focus on the hidden semi-Markov model (HSMM), used in Liang et al. [5] for the task of aligning segments of text to records in a knowledge base without supervision.

      iv. As in Liang et al. [5], we are interested in learning a generative model of text so that we can minimize the amount of supervision necessary for training an information extraction system.

      v. Also that although worse sample complexity, using an approximate posterior with monte carlo sampling achieves comparable performance.

2. Background and Notation

   (a) Formal notation for elements of the dataset

   (b) Define the distribution we would like to learn: $p(z \mid y, x)$. z and x are placeholders and will change, but y is always the text.

(c) Link to rotowire example (argument is that ACE is made up of ontonotes-like sentences, so all short-form)

(d) Clarify that the scope of posterior inference is very general.

3. Proposal

   (a) Outline approach

      i. Choose a subset of available data as conditioning, and thus it is not modelled.

      ii. The joint distribution of the remaining variables, both observed and unobserved, will be modelled.

   (b) Link back to motivation. We want to scale information extraction by requiring less supervision.

   (c) We present one model as an example, then later demonstrate how the framework can handle extensions of the model.

   (d) Define generative model: HSMM as in [5], and [7]. The generative story (a picture would be helpful):

      i. Fill in values

      ii. Choose alignments

      iii. Choose words

   (e) Define IE as the distribution we would like to learn

      i. Values: $p(c, \mathbf{v} \mid \mathbf{y}, \mathbf{e}, \mathbf{t})$ (Just this one)

   (f) We either use the posterior distribution of the conditional model or learn an approximation of it.

   (g) Argue that the segmental model encourages more coverage by adding pairwise dependencies between labelings [5]. We will check how much structure in the generative model aids information extraction.

   (h) Training and Inference

      i. As we are dealing with documents of significant length, we train with an approximate posterior in order to satisfy memory constraints.

      ii. Highlight that the approx posterior is a SEPARATE model that can be used completely independently from generative model, i.e. we throw away generative model after training.

      iii. We maximize a lower bound on the log marginal likelihood, called the evidence lower bound.

   (i) Extension, `Types`

      i. Introduce new step in generative process

      ii. Learn a boolean function that composes predicates applied to existing records

      iii. The search space is very large, so we must either constrain our model in a very clever way or obtain large amounts of data as any stochastic gradient estimator will have very high variance

   (j) Extensions

      i. learn new types as functions of existing ones

    ii. learn a randomly initialized embedding of the type and a neural network directly to predict the value
    iii. let the input distribution be

(k) Experiments, evaluation, and expectation

    i. Evaluate on Rotowire? Highlight long-form text
    ii. Also on ACE
    iii. As corpora may be too large, we might need more hierarchy in the generative model
    iv. and also since memory is linear in the length of the sequence, we may have to resort to approximate inference. We can optimize a lower bound on the marginal likelihood with an approximation of the posterior distribution [1].
    v. Evaluation metrics: for slot-filling, we evaluate the

(l) Conclusion

    i. Please accept!
    ii. Recap: Minimal supervision IE systems so that they can scale to extracting information for large information networks from large bodies of text.

References

[1] Yuntian Deng, Yoon Kim, Justin Chiu, Demi Guo, and Alexander M. Rush. Latent alignment and variational attention. *CoRR*, abs/1807.03756, 2018. URL http://arxiv.org/abs/1807.03756.

[2] T. George Doddington, C. Alexis Mitchell, T. Mark Przybocki, N. Lance Ramshaw, C. Stephanie Strassel, and N. Ralph Weischede. The automatic content extraction (ACE) program–tasks, data, and evaluation. 2004. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.78.8442&#38;rep=rep1&#38;type=pdf.

[3] Aria Haghighi and Dan Klein. Coreference resolution in a modular, entity-centered model. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 385–393, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. ISBN 1-932432-65-5. URL http://dl.acm.org/citation.cfm?id=1857999.1858060.

[4] Kenton Lee, Luheng He, and Luke Zettlemoyer. Higher-order coreference resolution with coarse-to-fine inference. *CoRR*, abs/1804.05392, 2018. URL http://arxiv.org/abs/1804.05392.

[5] Percy Liang, Michael I. Jordan, and Dan Klein. Learning semantic correspondences with less supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, pages 91–99, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-45-9. URL http://dl.acm.org/citation.cfm?id=1687878.1687893.

[6] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 455–465, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=2390948.2391003.

[7] Sam Wiseman. Learning neural templates for text generation. *CoRR*, abs/lol, 2018. URL `http://arxiv.org/abs/lol`.

[8] Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. Challenges in data-to-document generation. *CoRR*, abs/1707.08052, 2017.

[9] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45. Association for Computational Linguistics, 2017. doi: 10.18653/v1/D17-1004. URL `http://aclweb.org/anthology/D17-1004`.