Information Extraction with Weak Supervision

Keywords information networks, natural language processing, information extraction, latent variable models

Relevance to BAA As information networks get larger and more complex, acquiring explicit supervision for the training of information extraction systems becomes extremely expensive. Information networks are also dynamic; over time one method for representing information may become inadequate. Currently, both the representation of the information in a knowledge base as well as the extraction process itself must be hand-designed. In this proposal we present a framework towards automating the training of information extraction systems with minimal supervision.

Introduction Natural language processing contains two separate but closely related subfields: natural language understanding (NLU) and natural language generation (NLG). Recent approaches to information extraction use only the NLU perspective and frame extraction as a classification problem. We argue that the two perspectives, NLU and NLG, are complementary and capitalize on their duality by proposing a method to train a NLU system without direct supervision. More specifically, we train an information extraction system by using the performance of a deep generative model as signal. By training in this fashion we obtain an information extraction system that extracts facts that best explain the given text.

Recent work using neural network-based systems cast information extraction as a supervised problem [2, 6]. Although approaches do incorporate structure (?) into internal representations, for example to integrate multiple sources of information [4], the final output still uses strong supervision during training. Finding strong supervision is a difficult task, and as datasets get larger we must approach information extraction from a different perspective. Latent variable models (LVMs) are one method for alleviating the need for supervision. LVMs do not require manual labels, as they instead treat quantities of interest as latent random variables and deal with them in a probabilistically principled fashion. We will leverage recent advances in deep generative modeling in order to train without explicit supervision.

Neural models for language generation have seen much progress in recent years, with state of the art performance in both language modeling and translation [**?**] (MoS + Transformer). By integrating the powerful language modeling capabilities of neural networks with the inductive biases of graphical models through LVMs, we can leverage the LVM framework to derive a principled method for training an information extraction system with less supervision. Namely, we turn to an efficient technique for training hard attention that relies on variational inference [1]. This technique is applicable to sequential LVMs such as hidden Markov models and hidden semi-Markov models (HSMMs). HSMMs have been applied to small scale datasets for text generation [5] as well as without integrating neural networks [3]. We will scale sequential LVMs to large datasets through variational inference and use the signal from our generative model in order to train an information extraction system.

Background We consider datasets consisting of aligned data and text $\left\{(\mathbf{r}^{(1)}, \mathbf{y}^{(1)}), (\mathbf{r}^{(2)}, \mathbf{y}^{(2)}) \dots\right\}$. For brevity, we refer to a single datum and text as $\mathbf{r}, \mathbf{y}$, omitting the superscript. Each datum $\mathbf{r} = \{r_1, \dots, r_N\}$ is a set of N records, each of which has an entity, type, and value $r_i = (e_i, t_i, v_i)$. The datum $\mathbf{r}$ is a flattened representation of an information network. Whereas an information network may also be represented as a hypergraph, $\mathbf{r}$ is a list of relations or 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.

Let random 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 [6]. 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 $\mathbf{x} = \{e_1\}$. (Will switch to ACE dataset after I figure out what's going on with the data.)

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 $a_t = i$.

Proposal We propose to demonstrate the efficacy of the LVM framework in the weakly supervised information extraction setting. 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. The conditional model 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 $v_i \sim \text{Cat}(f_{t_i}(e_i))$, where f_{t_i} is a neural network shared across record types.

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. Each record alignment a_i points to the index of its corresponding record. The records 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.

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 a record: $y_{ij} \sim \text{Cat}(f_\theta(\mathbf{y}_{i<j}, a_i, \mathbf{v}, \mathbf{e}, \mathbf{t})) = \text{Cat}(f_\theta(\mathbf{y}_{i<j}, v_i, e_i, t_i))$, where f_θ is another neural network.

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. Characterized by large graphs and voluminous data.
      ii. Information networks must also represent knowledge wihich is both uncertain as well as dynamic; over time one method for representing information may become inadequate.
      iii. Obtaining supervision for large networks is expensive, especially if the underlying schema is subject to change.
      iv. TODO: Make expensive argument more concrete by hammering in the dynamic schema aspect. This argument might be weak since the model would also have to be retrained, but that's much cheaper than additional labeling.
      v. TODO: possibly establish a more military-focused or at least information-gathering running example, as opposed to basketball.
      vi. In this proposal we present a framework towards automating the training of information extraction systems with minimal supervision.
   (b) Decomposition of NLP into NLU and NLG (Keep this paragraph high-level, just an introduction to the thesis and proposal.)
      i. NLP consists of two separate but closely related subfields: natural language understanding (NLU) and natural language generation (NLG).
      ii. Although both subfields have been dominated by neural network-base models, there is a striking difference.
      iii. NLU models capitalize primarily on the representational power of neural networks. Rather than requiring features as input, neural networks learn representations of the data automatically. (Maybe move the whole NLU paragraph here, but probably not since I feel like I have a strong opening paragraph.)
      iv. However, NLG utilizes neural networks not only for representation learning but also for generative modeling. Success on tasks such as language modeling and machine translation demonstrate that generative and conditional models parameterized by neural networks are able to approximate not only simple distributions over labels, such as entity or not, but also complex joint distributions. (Maybe move the whole generative paragraph here, X)
      v. Recent approaches to information extraction use only the NLU perspective and frame extraction as a classification problem, and thus do not take advantage of the generative capability of neural networks.
      vi. TODO: Possibly give examples QA, information extraction (dos santos + sam).
      vii. We argue that the two subfields' approaches are complementary, and capitalize on their duality by proposing a method to train a NLU system without direct supervision.
   (c) Supervision in NLU (Need to hammer in the framing of supervision)
      i. Recent work using neural network-based systems cast information extraction as a supervised problem [2, 6].

ii. Even with structured representations [4], the final output is still supervised.
iii. Again, supervision does not scale.
iv. Latent variable models (LVMs) are one method for alleviating the need for supervision.
v. LVMs do not require manual labels, as they instead treat quantities of interest as latent random variables and deal with them in a probabilistically principled fashion.

(d) Recent advances in NLG (Need to argue the benefits of generative modeling, is the recent success enough?)
i. Deep generative models, namely LVMs with neural network components, have integrated the flexibility of neural networks with the inductive biases of graphical models.
ii. Most importantly, efficient techniques for training hard attention that rely on variational inference [1]. This technique is applicable to sequential LVMs such as hidden Markov models and hidden semi-Markov models (HSMMs).
iii. Use HSMM as signal for training IE.

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) Define generative model: HSMM as in [3], and [5]. The generative story (a picture would be helpful):
i. Fill in values
ii. Choose alignments
iii. Choose words
(b) 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)
(c) 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. We maximize a lower bound on the log marginal likelihood, called the evidence lower bound.
(d) Experiments
i.
(e) Conclusion
i.

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] Cícero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. Classifying relations by ranking with convolutional neural networks. *CoRR*, abs/1504.06580, 2015. URL `http://arxiv.org/abs/1504.06580`.

[3] 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`.

[4] Dirk Weissenborn, Tomas Kocisky, and Chris Dyer. Dynamic integration of background knowledge in neural nlu systems. *CoRR*, abs/1706.02596, 2017. URL `http://arxiv.org/abs/1706.02596`.

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

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