Information Extraction with Weak Supervision

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

Relevance to Army BAA: II. A. c. iii. (3) Information Networks In order to provide decision makers with the information necessary to make informed choices as well as predict the effects of those choices, we must have an efficient representation of relevant information and a predictive model for the resultant state of the world given a choice. Information networks provide a graphical representation of information and how it propagates through a network. We focus on *knowledge graphs*, information networks where each node contains a set of facts about an entity and an edge describes how the facts in one node influence the facts in another. Knowledge graphs provide an intuitive and queryable representation of knowledge. A decision maker may query the relevant nodes to gain situational awareness, and when simulating a decision that alters the information in one node the graph can easily propagate those changes by virtue of its representation.

Given that a knowledge graph must represent an extremely large number of facts and relationships, it is infeasible to specify these completely by hand. Many recent works have focused on learning to fill in the missing edges of a knowledge graph by recognizing patterns in fully-labeled subgraphs in order to predict whether there should be an edge between two nodes in an unlabeled subgraph [2]. By filling in missing edges, we are able to use the relationships specified by the edges in order to reason about the facts contained in a node conditioned on the facts of its neighbours. For example, given two nodes corresponding to two different entities, we can leverage the relationships between the two nodes to infer the values of one given the other.

However, in addition to supervision over subgraphs, we generally also have access to large amounts of unlabeled and unstructured data in the form of natural language text. We propose an approach that is orthogonal to edge completion, which instead relies on jointly modeling text and the nodes of the information network. We aim to leverage natural language text by learning an information extraction system that is able to extract facts from text to fill in the nodes of a knowledge graph given **only** the assumption that the text was generated from the information represented in the knowledge graph. Although this is a strong assumption, we believe it is a reasonable starting point. We note that information extraction is complementary to edge completion as a method for inferring missing values in the nodes of a knowledge graph, and although we focus on information extraction due to its ability to leverage text, the two approaches can be combined in future work.

In this proposal we present a method towards automating the training of information extraction systems with unlabeled corpora by recasting the information extraction problem as a conditional generative modeling problem. Specifically, we plan to use the performance of a deep generative model of text as signal for learning an extraction system.

Introduction The goal of information extraction is to produce structured representations of information given unstructured text. We use summaries of basketball games as an example to ground the concept of an information extraction system. In the context of a basketball game summary, an information extraction system would infer all the statistics associated with a player given the summary. A typical approach to an information extraction system is the following pipeline, where each stage has a model that is trained independently from the other stages: first segment the text into entities and values, then extract named entities and possibly perform coreference resolution by predicting whether segments refer to the same entity, and finally relation extraction where we identify the relationships between the extracted entities and values. We can then use the extracted relationships along with the associated entities and values to populate the nodes of a knowledge graph, where each node would correspond to a player and contain their associated statistics. See Figure 1 for an example of this process. This process is orthogonal and complementary to using existing nodes in a knowledge graph to fill in missing ones. As we are interested in utilizing large amounts of unlabeled corpora for the training of an information extraction system, we restrict our focus to learning the values of each node independently given the text. Neural networks, in combination with latent variable models (LVMs), provide a method for training such an information extraction pipeline end-to-end.

The progression from a pipelined system into one that is trained jointly has precedent in the field of natural language processing: machine translation. Previously, statistical machine translation utilized a highly pipelined approach, where each stage utilized a model that was trained independently of the others. The approach was unified with an end-to-end neural model in Bahdanau et al. [1]. We recently recast Bahdanau et al. [1]'s model in the LVM framework in Deng et al. [3] which resulted in performance gains as well as improved sample complexity. Our proposal is inspired by the improvements seen in machine translation: we wish to specify an information extraction system that is learned end-to-end rather than stage by stage, and we use the LVM framework to do so. LVMs provide a principled way of specifying semi-supervised models [6], and have been shown to have better sample complexity than discriminative models [3, 8].

Although LVMs have recently been proposed in the context of knowledge graph completion [2, 9], they either do not utilize text or do not have a (conditional) generative model of text. We argue that the generative model we specify should be as close to the data generating process as possible. In particular, Chen et al. [2] formalize predicting the relationship between two nodes as a LVM but do not utilize text at all. Qu et al. [9] also learn a LVM for the relationships between nodes, which includes using the text to learn a distributed representations of entities. However, their approach does not take into account the generative process of the text, which limits the expressibility of their model. We propose to explicitly model how a text is created given a knowledge graph using a conditional generative model with latent variables.

In this proposal, we focus on the class of LVMs known as hidden semi-Markov models (HSMMs), used in Liang et al. [7] as a conditional generative model for the task of aligning segments of text to nodes in a knowledge graph without supervision. As in Liang et al. [7], we are interested in learning a conditional generative model of text so that we can minimize the amount of supervision necessary for training an information extraction system. The performance of their conditional 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 a semi-supervised information extraction which aims to model not just the alignments from segments of text to nodes in a knowledge graph but also the values in the nodes themselves. By parameterizing the HSMM with neural networks as in Wiseman [12], 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.

Background We consider datasets consisting of aligned data and text $\left\{(\mathbf{r}^{(1)}, \mathbf{y}^{(1)}), (\mathbf{r}^{(2)}, \mathbf{y}^{(2)}) \ldots\right\}$. For brevity, we refer to a single datum and text as $\mathbf{r}, \mathbf{y}$, omitting the superscript. Each datum $\mathbf{r} = \{r_1, \ldots, r_N\}$ 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 vocabulary V.

We proceed to detail how to specify a LVM, then provide a concrete example linking the above dataset description to our LVM formulation: 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 [13]. 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 value v_1 given the entity Jeremy Lin, the type POINTS, as well as the text $\mathbf{y}$. In this case, we would have $\mathbf{z} = \{v_1\}$ and $\mathbf{x} = \{e_1, t_1\}$. 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 $\mathbf{z} = \{v_1, t_1\}$ and $\mathbf{x} = \{e_1\}$.

Proposal We propose to verify the efficacy of the LVM framework in the semi-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 proposed LVM is a conditional generative 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 [12] and Liang et al. [7], 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. For our model, we have observed $\mathbf{x} = \{\mathbf{e}, \mathbf{t}\}$ and latent $\mathbf{z} = \{\mathbf{v}, \mathbf{a}\}$, where $\mathbf{e}$ are the entities, $\mathbf{t}$ are the types, $\mathbf{v}$ are the values, and $\mathbf{a}$ are record indices. `Values` is given by the following generative process:

1. Prior Value Choice: $p(\mathbf{v} \mid \mathbf{e}, \mathbf{t}) = p(\mathbf{v} \mid \mathbf{x})$. For each entity and type pair in our datum of records, we predict a value. For example, given 'Jeremy_Lin' and POINTS, we predict 19. Each value is predicted independently. As both the entites and types are observed, we have $\mathbf{x} = \{\mathbf{e}, \mathbf{t}\}$.

2. Prior Record Choice: $p(\mathbf{a} \mid \mathbf{v}, \mathbf{x}) = \prod_i p(a_i \mid a_{i-1}, \mathbf{v}, \mathbf{x})$. Conditioned on our choices of values as well as the given entities and records, in other words a completed data with no missing values, we choose a sequence of records $\mathbf{a} = \{a_1, \ldots, a_I\}$ to describe.

3. Word Choice: $p(\mathbf{y} \mid \mathbf{a}, \mathbf{v}, \mathbf{x}) = p(\mathbf{y} \mid \mathbf{z}, \mathbf{x})$. For each chosen record a_i, we choose a sequence of words $\mathbf{y}_i = \{y_{i1}, \ldots, y_{iJ}\}$ to describe the record. With the HSMM formulation, we have that within each segment aligned to a record, the words are modeled autoregressively. This distribution is the likelihood of the text given the latent variables $\mathbf{z}$, the record choices and value choices.

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. [3], where we learn an approximation of the posterior distribution $q(\mathbf{z} \mid \mathbf{y}, \mathbf{x})$ with a separate model. We can train the conditional generative model and the approximate posterior jointly by maximizing a lower bound on the log marginal likelihood or evidence of $\mathbf{y}$, called the evidence lower bound (ELBO) 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.

We will evaluate our approach on the TAC KBP 2015 slot filling, TACRED [14], and ROTOWIRE [13] datasets in order to compare to previous work. The goal will be to demonstrate competitive performance on extraction metrics such as precision, recall, and F1, while using as little supervision as possible by ignoring subsets of the given information. For example, with the model `Values`, we can allow the model to only learn from subsets of the given values (or none at all) in ROTOWIRE, such as only the home team's statistics, at training time. Given success in that goal, the next step would be to extend the model with more structure with the aim of boosting sample and label efficiency. Possible extensions in this direction include parameterizing entities including entity tracking and coreference resolution [4]; learning the types of relations in a semi-supervised manner; leveraging discourse structure to improve the conditional generative model [11]; explicitly modeling nuisance variables such as author style [5]; and incorporating multi-hop reasoning in order to leverage relationships between entities [2, 10].

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. Importance of representation and correctness of knowledge in decision making, as well as the power of prediction.
        - ii. Information networks are one way of representing knowledge as a graph as well as interactions?
        - iii. What are information networks? Graphical representation of objects, their characteristics of interest, and their relationship to objects.
        - iv. Recent work focuses on learning structure between nodes in order to fill parts of the graph that may be missing [2].
        - v. We focus on the orthogonal approach of completing knowledge graphs using large amounts of unstructured text rather than the relationships between nodes.
        - vi. 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).
        - vii. In this proposal we present a framework towards automating the training of text-centric information extraction systems with minimal supervision.
        - viii. Shine some hope on interventional distribution representations? (This might be a little difficult)
    - (b) Supervision in Information Extraction
        - i. Define information extraction
            - A. The goal is to produce structured representations of information from a given unstructured text.
            - B. Ideally, but not necessarily computer-readable in addition to human-readable.
            - C. A typical pipeline for information extraction includes text segmentation, named entity recognition, coreference resolution, relation extraction, and finally producing structured representations of the unlabeled text.
            - D. This is knowledge-base completion.
            - E. We are primarily interested in knowledge-base completion, as the other tasks such named entity recognition are typically part of a pipeline aimed at knowledge-base completion.
        - ii. Argue for LVM approach to unify all parts of the pipeline and train end-to-end with minimal supervision.
        - iii. We aim to unify these two approaches through recently developed techniques for training LVMs parameterized with neural networks.
    - (c) End-to-end trends and LVMs
        - i. Draw parallels to progress in translation, mainly in terms of structure?
            - A. Statistical machine translation: pipelines
            - B. End-to-end with Sutskever

C. Deterministic structure via attention Bahdanau

D. Latent attention Deng et al. [3]

ii. Leverage modularity of both the neural network and LVM frameworks to incorporate more of the pipeline into a joint training framework.

iii. Qu et al. [9] Semi-supervised relation extraction (which is pretty much similar) How is what I want different? Capitalize on the progress of powerful generative models, can argue that their model is much weaker in terms of generative power, but the framework is similar (they perform coordinate ascent in a LVM).

(d) Recent advances in neural LVMs

i. Semi-supervised LVMs Kingma et al. [6]?

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

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

iv. In this proposal, we focus on the hidden semi-Markov model (HSMM), used in Liang et al. [7] for the task of aligning segments of text to records in a knowledge base without supervision.

v. As in Liang et al. [7], 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.

vi. 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, which we will serve as a starting point for the proposed research.

(d) Define generative model: HSMM as in [7], and [12]. 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) 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.

(h) Experiments, evaluation, and expectation

i. Evaluate on Rotowire. Highlight long-form text?

ii. We evaluate `Values` using the precision, recall, and F1 score on the task of predicting the values associated with entities, otherwise known as slot-filling.

iii. What would success look like?

A. Competitive to supervised methods when supervision is available

B. But able to be applied when supervision is not available

C. Able to leverage lots of unlabeled data during training, and success would see a marked improvement over purely supervised methods as well as the purely supervised version of this model.

D. Would provide explanations for the answers (ie segmentations).

iv. Also on ACE?

(i) Future work

i. Incorporate more structure into the generative model, for example entity tracking or coreference resolution Haghighi and Klein [4].

ii. Model more structure in the data, for example the edges between nodes in the knowledge graph Chen et al. [2].

iii. ‘Multi-hop’ reasoning, where we try to compose relationships to infer new ones, i.e. through unification Chen et al. [2], Rocktäschel and Riedel [10].

(j) 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] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. URL `http://arxiv.org/abs/1409.0473`.

[2] Wenhu Chen, Wenhan Xiong, Xifeng Yan, and William Yang Wang. Variational knowledge graph reasoning. *CoRR*, abs/1803.06581, 2018. URL `http://arxiv.org/abs/1803.06581`.

[3] 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`.

[4] 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`.

[5] Wei-Ning Hsu, Yu Zhang, and James R. Glass. Learning latent representations for speech generation and transformation. *CoRR*, abs/1704.04222, 2017. URL `http://arxiv.org/abs/1704.04222`.

[6] Diederik P. Kingma, Danilo Jimenez Rezende, Shakir Mohamed, and Max Welling. Semi-supervised learning with deep generative models. *CoRR*, abs/1406.5298, 2014. URL `http://arxiv.org/abs/1406.5298`.

[7] 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`.

[8] Andrew Y. Ng and Michael I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, pages 841–848, Cambridge, MA, USA, 2001. MIT Press. URL `http://dl.acm.org/citation.cfm?id=2980539.2980648`.

[9] Meng Qu, Xiang Ren, Yu Zhang, and Jiawei Han. Overcoming limited supervision in relation extraction: A pattern-enhanced distributional representation approach. *CoRR*, abs/1711.03226, 2017. URL `http://arxiv.org/abs/1711.03226`.

[10] Tim Rocktäschel and Sebastian Riedel. End-to-end differentiable proving. *CoRR*, abs/1705.11040, 2017. URL `http://arxiv.org/abs/1705.11040`.

[11] Christina Sauper and Regina Barzilay. Automatically generating wikipedia articles: A structure-aware approach. 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 208–216, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-45-9. URL `http://dl.acm.org/citation.cfm?id=1687878.1687909`.

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

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

[14] 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`.