Unsupervised Gaussian embedding matching using normalising flows

David Yenicelik (ETHZ) - Master Thesis Proposal

Motivation

In Natural Language Processing (NLP), bilingual lexical induction (BLI) is a problem of inferring word-to-word mapping between two languages. While supervised BLI may be learned trivially from a dictionary, unsupervised BLI is highly non-trivial, and serves as a backbone to many unsupervised Neural Machine Translation (NMT) systems, without which the overall MT performance drastically drops [1] [2]. In addition to the unsupervised setting, words in a source language A often do not carry a one-to-one correspondence with words in a target language B. This further increases the difficulty of finding a (bijective) mapping between the two embedding spaces.

Background

The above two problems can be regarded as finding an invertible transformation from one (embedding) space $\mathcal{X} \in \mathbf{R}^d$ into another $\hat{\mathcal{X}} \in \mathbf{R}^d$. Normalising flows [3] [4] have proven to be a powerful tool in modeling such relations. As such, the aim of this project is to find a model based on normalising flows which is able to find an an invertible mapping f from $\mathcal{X}$ to $\hat{\mathcal{X}}$. To keep the discussion focused, this thesis deals with the problem of finding a model for bilingual lexicon matching.

Gaussian Embeddings

Gaussian embedding [5] is a set of probability distributions which embed tokens $v_i = \mathcal{N}(x; \mu_i, \Sigma_i)$ as Gaussian distributions in the latent embedding space with learnable parameters $\theta = [\mu_i, \Sigma_i]$. This is in contrast to the more commonly used vector-embeddings, as proposed by word2vec [6] or GloVe [7], where the learned embedding is merely a point vector $v_i \in \mathcal{R}^d$. Vilsnis et al., [5] report that the Gaussian embeddings are better at capturing uncertainty about a representation and its relationships, but are also more difficult to train.

Normalising Flows

Normalising flows [3], [8] are a statistical technique where a series of invertible transformations f_t are applied to a simple distribution $z_0 \sim q_0(z)$, to yield increasingly complex distributions $z_t = f_t(z_{t-1})$, s.t. the last iterate z_T has the desired and more flexible distribution. As long as we can efficiently compute the Jacobian determinant of the transformation bijection f_t, we can both (1) evaluate the density of our data (by applying an inverse transformation and computing the density in the base distribution), and (2) sample from our complex distribution (by sampling from the base distribution and applying the forward transformation) These can be used for classification and clustering [8], [variational inference tasks [9] such as image-generation [4]], enriching the posterior (and prior!) [3], and density estimation [10].

Unsupervised Bilingual Lexicon Matching

Bilingual lexicon matching is the task of finding a target token in language B, a corresponding source token in language A. More generally, this can be seen as translating one vector-space into another, addressing the need for a common embedding space amongst items [6], [11]. Existing work includes cross-domain audio/text/image generation [12] [13] [14] [15], and also unsupervised language translation [16]. There has also been some work in matching tokens of a source language with bilingual lexicon matching [17] [18], [19]. Unsupervised methods aim at minimizing a global cosine error between the two embedding spaces [20].

Scope Of Work

Continuing where [21] left off, we want to investigate the performance of using normalising flows to model unsupervised lexicon matching between two probability distributions, which are defined by Gaussian emebddings, which have a one-to-one correspondence to tokens in the respective languages. We aim to use Gaussian embeddings for the robustness, and better integration into the probabilistic perspective. The method discussed in the paper also relies a lot on the (semi-)supervised loss component. We aim to investigate why this is the case, and would like to revise a method which is more robust in the fully unsupervised setting.

Minimal goals: We propose the following steps on achieving this goal.

1. Replicate the algorithms and models which can generate through Gaussian embedding [5]. Do one sanity check by doing a sanity check on one of (SimLex / WordSim)

2. Replicate "Density matching for bilingual word embedding" to setup a baseline normalising flow between vector-word-embeddings [21].

3. Define loss between predicted and target embeddings (if change in definition is necessary)

4. Change the embeddings in point 2. to use Gaussian embeddings. Implement Loss functions found in point 3.

Extended goals: If the above points provide good performance , we would like to expand on the below points.

1. Implement a fully unsupervised extension by investigating the shortcomings of [21].

2. Investigate "deeper" normalising flows than the linear flow in [21] as such [22] [23].

Contingency Plan / Further extended goals: Implementing the following points would allow for an applied perspective of this approach, showing that this methodology allows for more robust mapping, also outside the field of NLP.

1. Train embeddings for job-systems ESCO and AUGOV using Skip-Gram or Co-Occurence-matrix based. Do a sanity check.

2. Train Gaussian embeddings for job-systems. Do a sanity check.

3. Generate a small validation dataset between the European- and Australian job-system.

4. Setup some baseline algorithms based on NLP, graph-matching, colinear-PCA for matching as a non-NLP benchmark environment. Compare against above-proposed methods.

5. Find a normlising flow model to transform one job system into another.

References

[1] G. Lample, A. Conneau, L. Denoyer, and R. M., "Joint training for neural machine translation," 2018.

[2] M. Artetxe, G. Labaka, A. E., and K. Cho, "Unsupervised neural machine translaton," 2018.

[3] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," 2015.

[4] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," 2017.

[5] L. Vilnis and A. McCallum, "Word representations via gaussian embedding," 2015.

[6] M. T., C. K., G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[7] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.

[8] E. Tabak and E. Vanden-Eijnden, "Density estimation by dual ascent of the log-likelihood," 2010.

[9] D. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," 2017.

[10] D. Kingma and P. Dhariwal, "Glow : Generative flow with invertible 1 1 convolutions," 2018.

[11] S. Kudugunta, A. Bapna, I. Caswell, N. Arivazhagan, and O. Firat, "Investigating multilingual nmt representations at scale," 2019.

[12] S. Ma, D. Mcduff, and Y. Song, "M 3 d-gan: Multi-modal multi-domain translation with universal attention," 2019.

[13] X. Chen, C. Xu, X. Yang, and D. Tao, "Attention-gan for object transfiguration in wild images," 2018.

[14] J. Lin, Y. Xia, T. Qin, Z. Chen, and T. Liu, "Conditional image-to-image translation," 2018.

[15] J. Zhu, T. Park, P. Isola, and A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," 2017.

[16] G. Lample, M. Ott, A. Conneau, L. Denoyer, and M. Ranzato, "Phrase-based amp; neural unsupervised machine translation," 2018.

[17] M. Artetxe, G. Labaka, and E. Agirre, "A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings," 2018.

[18] M. .Artetxe, G. Labaka, and E. Agirre, "Bilingual lexicon induction through unsupervised machine translation," 2019.

[19] Y. Cheng, p. 28, 2017.

[20] A. Conneau, G. Lample, A. Ranzato, L. Denoyer, and H. Jégou, "Word translation without parallel data," 2018.

[21] C. Zhou, X. Ma, D. Wang, and G. Neubig, “Density matching for bilingual word embedding,” 2019.

[22] J. Ho, X. Chen, A. Srinivas, Y. Duan, and P. Abbeel, “Flow++: Improving flow-based generative models with variational dequantization and architecture design,” 2019.

[23] C. Durkan, A. Bekasov, and G. Murray, I. Papamakarios, “Neural spline flow,” 2019.

[24] J. Agnellit, M. Cadeiras, E. Tabak, C. Turnert, and E. Vanden-Eijnden, “Clustering and classification through normalizing flows in feature space,” 2010.