Feat2Vec: Supplemental Materials

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Supplemental proof for Feat2Vec paper

0.1 Proof to Theorem 1

Theorem 1. *The gradient for learning embeddings with* Feat2Vec *is a convex combination of the gradient from* n *targeted Factorization Machines for each feature in the data when each feature group is a singleton, where* n *is the total number of features in the dataset.*

Proof. Let $S^+_{\kappa_i}$ denote the positively labeled records whose corresponding negative samples resample feature κ_i. For convenience, suppress the inclusion of learned parameters θ in the notation in this section while understanding the feature extraction functions $\vec{\phi}$ implicitly include these parameters. We can express the loss function $L(.)$, the binary cross-entropy of the data given the $\mathrm{Feat2Vec}$ model, as follows:

$$
\begin{aligned}
L(S^+|\vec{\phi}) =& \frac{1}{|S^+|} \sum_{\vec{x}^+ \in S^+} \Big(\log(\tilde{p}(y=1|\vec{\phi}, \vec{x}^+)) + \sum_{\vec{x}^- \sim \mathcal{Q}(.|\vec{x}^+)}^{k} \log(\tilde{p}(y=0|\vec{\phi}, \vec{x}^-)) \Big) \\
=& \frac{1}{|S^+|} \sum_{\vec{x}^+ \in S^+} \Big(\log(\tilde{p}(y=1|\vec{\phi}, \vec{x}^+, \vec{x}^+ \in S^+_{\kappa_i}) p(\vec{x}^+ \in S^+_{\kappa_i})) \\
&+ \sum_{\vec{x}^- \sim \mathcal{Q}(.|\vec{x}^+)}^{k} \log(\tilde{p}(y=0|\vec{\phi}, \vec{x}^-, \vec{x}^+ \in S^+_{\kappa_i}) p(\vec{x}^+ \in S^+_{\kappa_i})) \Big) \\
=& \frac{1}{|S^+|} \sum_{i=1}^{n} \sum_{\vec{x}^+ \in S^+_{\kappa_i}} \Big(\log(\frac{e^{s(\vec{x}^+, \vec{\phi})} p(\vec{x}^+ \in S^+_{\kappa_i})}{e^{s(\vec{x}^+, \vec{\phi})} + P_{\mathcal{Q}}(\vec{x}^+|\vec{x}^+, \vec{x}^+ \in S^+_{\kappa_i})}) \\
&+ \sum_{\vec{x}^- \sim \mathcal{Q}(.|\vec{x}^+, \vec{x}^+ \in S^+_{\kappa_i})}^{k} \log(\frac{P_{\mathcal{Q}}(\vec{x}^-|\vec{x}^+, \vec{x}^+ \in S^+_{\kappa_i}) p(\vec{x}^+ \in S^+_{\kappa_i})}{e^{s(\vec{x}^-, \vec{\phi})} + P_{\mathcal{Q}}(\vec{x}^-|\vec{x}^+, \vec{x}^+ \in S^+_{\kappa_i})}) \Big)
\end{aligned}
$$

Submitted to 32nd Conference on Neural Information Processing Systems (NIPS 2018). Do not distribute.

Note now that $P_{\mathcal{Q}}(\vec{x}|\vec{x}^+, \vec{x}^+ \in S^+_{\kappa_i})$ is simply the probability of the record's feature value $\vec{x}_f$ under the second step noise distribution $\mathcal{Q}_2(\mathrm{X_f}, \alpha_2)$: $P_{\mathcal{Q}}(\vec{x}|\vec{x}^+, \vec{x}^+ \in S^+_{\kappa_i}) = P_{\mathcal{Q}_2}(\vec{x}_f)$

$$
\begin{aligned}
=&\frac{1}{|S^+|}\sum_{i=1}^{n}\sum_{\vec{x}^+\in S^+_{\kappa_i}}\Big(\log(\frac{e^{s(\vec{x}^+,\vec{\phi})}p(\vec{x}^+\in S^+_{\kappa_i})}{e^{s(\vec{x}^+,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^+_{\kappa_i})})+\sum_{\vec{x}^-\sim\mathcal{Q}(.|\vec{x}^+,i\in S^+_{\kappa_i})}^{k}\log(\frac{P_{\mathcal{Q}_2}(\vec{x}^-_f)p(\vec{x}^+\in S^+_{\kappa_i})}{e^{s(\vec{x}^-,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^-_f))}\Big)\\
=&\frac{1}{|S^+|}\sum_{i=1}^{n}\sum_{\vec{x}^+\in S^+_{\kappa_i}}\Big(\log(\frac{e^{s(\vec{x}^+,\vec{\phi})}}{e^{s(\vec{x}^+,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^+_{\kappa_i})})+\log(p(\vec{x}^+\in S^+_{\kappa_i})^{k+1})\\
&+\sum_{\vec{x}^-\sim\mathcal{Q}(.|\vec{x}^+,\vec{x}^+\in S^+_{\kappa_i})}^{k}\log(\frac{P_{\mathcal{Q}_2}(\vec{x}^-_f)}{e^{s(\vec{x}^-,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^-_f)})\Big)
\end{aligned}
$$

We now drop the term containing the probability of assignment to a feature group $p(\vec{x}^+ \in S^+_{\kappa_i})$ since it is outside of the learned model parameters $\vec{\phi}$ and fixed in advance:

$$
\begin{aligned}
\propto&\frac{1}{|S^+|}\sum_{i=1}^{n}\sum_{\vec{x}^+\in S^+_{\kappa_i}}\Big(\log(\frac{e^{s(\vec{x}^+,\vec{\phi})}}{e^{s(\vec{x}^+,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^+_{\kappa_i})})+\sum_{\vec{x}^-\sim\mathcal{Q}(.|\vec{x}^+,\vec{x}^+\in S^+_{\kappa_i})}^{k}\log(\frac{P_{\mathcal{Q}_2}(\vec{x}^-_f)}{e^{s(\vec{x}^-,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^-_f)})\Big)\\
\xrightarrow[|S^+|\to\infty]{}&\sum_{i=1}^{n}p(\vec{x}^+\in S^+_{\kappa_i})E\Big[\log(\frac{e^{s(\vec{x}^+,\vec{\phi})}}{e^{s(\vec{x}^+,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^+_{\kappa_i})})+\sum_{\vec{x}^-\sim\mathcal{Q}(.|\vec{x}^+,\vec{x}^+\in S^+_{\kappa_i})}^{k}\log(\frac{P_{\mathcal{Q}_2}(\vec{x}^-_f)}{e^{s(\vec{x}^-,\vec{\phi})}+P_{\mathcal{Q}_2}(\vec{x}^-_f)})\Big]\\
=&\sum_{i=1}^{n}p(\vec{x}^+\in S^+_{\kappa_i})E\Big[L(\vec{x}|\vec{\phi},\text{target}=f)\Big]
\end{aligned}
$$

Thus, the loss function is just a convex combination of the loss functions of the targeted classifiers for each of the p features, and by extension so is the gradient since:

$$
\frac{\partial}{\partial\phi}\sum_{i=1}^{n}p(\vec{x}^+\in S^+_{\kappa_i})E\Big[L(\vec{x}|\vec{\phi},\text{target}=f)\Big]=\sum_{i=1}^{n}p(\vec{x}^+\in S^+_{\kappa_i})\frac{\partial}{\partial\phi}E\Big[L(\vec{x}|\vec{\phi},\text{target}=f)\Big]
$$

Thus the algorithm will, at each step, learn a convex combination of the gradient for a targeted classifier on feature f, with weights proportional to the feature group sampling probabilities in step 1 of the sampling algorithm. Note that if feature groups are not singletons, the gradient from unsupervised $\mathrm{Feat2Vec}$ will analogously be a convex combination of n gradients learned from supervised learning tasks on each of the n feature groups. □