Using text and embedding models to learn Character and Character Relationship embeddings

Anonymous ACL-IJCNLP submission

1 Introduction

2 Formulation

Let $S_1, \ldots, S_n$ be the sequence of n sentences/paragraphs forming a piece of literature (play, novel, script etc.). We will be using a pretrained model such as SENTENCE-BERT (S-BERT) (Reimers and Gurevevych, 2019) or SENT2VEC (Pagliardini et al., 2018) to obtain embeddings of these sentences/paragraphs. Let v_S denote the pre-trained sentence representation of a sentence S. Let $c_1, \ldots, c_k$ denote the k characters in the piece. We will denote the representation of the character c_i by the embedding C_i. We also select a set of words which either can indicate a specific quality of an individual character(like hero, villain, evil, friendly, religious) or indicate a specific quality of relationship between two characters(like friendly, hostile, mother, father, son, daughter etc.). Lets call this set of m words $\{w_1, \ldots, w_m\}$. We will use a matrix U to embed these words where the i^{th} row u_i represents the w_i. We will use a network f_{rel} to denote the relationship between two characters, e.g., $f_{rel}(C_i, C_j)$ represents a vector embodying the relationship between two characters C_i and C_j.

Now, we will use contextual relationship to formulate similarity score based relationships to train the character embeddings, relationship embeddings and the relationship network f_{rel}. Let c_i and c_j occur in a sentence/paragraph S_k. Then we would like the following dot-product scores to be high after training -

- $c_i^T v_{S_k}$ since the sentence S_k is talking about the character c_i it is highly like to contain information about the characteristics of c_i.
- $c_j^T v_{S_k}$ (same rationale as above)
- $f_{rel}(c_i, c_j)^T v_{S_k}$ since the sentence involves the two characters c_i and c_j it is highly like to contain an interaction between the two characters which might be indicative of the relationship between them.
- Let $w_l \in S_k$ then we also want $f_{rel}(c_i, c_j)^T u_l$ to be high as well as w_l is highly likely to denote the relationship between c_i and c_j.
- Similarly, we want to keep $c_i^T u_l$ and $c_j^T u_l$ to be high as well.

We can replace one of the vectors in the above dot-product formulations with a randomly chosen one to obtain negative dot product pairs where we would like to keep the dot product low and finally come up with a similar loss function as WORD2VEC.

References

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 528–540, New Orleans, Louisiana. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.