Evaluating Deep Learning Approaches for Character Identification in Multiparty Dialogues

Krishna Prasad Sankaranarayanan and Sree Harsha Ramesh
University of Massachusetts Amherst
Amherst MA, 01002, USA

1 Outline

Character identification is an entity linking task that identifies each mention as a certain character in multiparty dialogue where mentions are typically nominals referring to a person and entities maybe speakers themselves or even external characters. Identifying such mentions as real characters requires cross-document entity resolution, which makes this task challenging. This task involves coreference resolution which clusters together the mentions corresponding to the same referent followed by an entity linking stage where the clusters of mentions are mapped to their corresponding entities. Historically, coreference models have been trained on the NewsWire dataset which is not as rich in terms of the complexity of the coreferences as those in multiparty dialogues. As has been the norm on various natural language processing tasks, deep-learning models are the state-of-the-art in coreference resolution as well. However, coreference resolution systems have been shown not to handle dialogues as well ([2],[3]). This motivates us to extend and evaluate the existing coreference systems including rule-based, statistical and deep-learning based models, for the annotated TV Show transcripts dataset released as part of SemEval 2018 Task-4 ¹.

2 Related Work

Henry Y. Chen et al [4] proposes a deep learning approach to coreference resolution and entity linking for character identification. It introduces a new agglomerative convolutional neural network for returning mention and mention pair embeddings. Entity links are then mapped to each referent separately by cluster embeddings. This method takes into consideration, 20 labels viz. Top 9 characters and an unknown label. It emphasizes on the intuition that the coreference resolution accuracy depends upon the size of the clusters. The combined implementation of cluster and mention embeddings bettered the singular use of mention embeddings in terms of accuracy. Although, agglomerative CNN being an incremental feature approach has its fair share of advantages in terms of word embeddings, the existing approach still lacks the handling of plurals and collective nouns. This scope for overall improvement is aimed to be leveraged by us while looking for extending and evaluating the deep learning approaches.

¹ https://competitions.codalab.org/competitions/17310
Clark and Manning [5] introduces an entity centric system using mention pair models as features. Agglomerative clustering is used to build coreference chains formed by merging pairs of clusters at each step. A key aspect of any incremental coreference system is its local decisions. Using this to full advantage, costs are assigned to each action which are in turn trained using a cost-sensitive classification. For the ranking model, the current mention is matched to the candidate antecedents simultaneously competing with each other. The resultant prediction model depends upon the previous actions which violate the IID (independent and identically distributed) assumption of statistical learning. Hence, imitation learning is used to classify whether a particular action is the one the expert policy would take at the current state.

Kevin Clarke [6] proposes a new approach to coreference resolution using distributed word representations. An incremental coreference system is defined which acts as a feed forward neural network for mention clusters rather than mention pairs. The usage of mention pairs do not enforce transitivity and therefore rely only on local pairwise information to make coreference decisions. Mention clusters on the other hand facilitate previous coreference decisions to inform the latest ones. This is an extension to [2] system wherein features are created between mention clusters using the pairwise probabilities of the mention pair model. This is extended by consideration of all features from vector representations of mention pairs to produce cluster level features. The actual benefits of deep learning on coreference are the lack of hand engineered features. This is leveraged by Clarke in creating a simple feature set which outperforms state-of-the-art approaches.

Sam Wiseman et al [7] presents a mention ranking model for coreference resolution. It emphasizes on anaphoricity detection and antecedent ranking with respect to learning feature representations. The training model using backpropagation is preceded by a pre-training segment comprising of two tasks viz. anaphoricity detection and antecedent ranking. The mention ranking model is trained with the slack-rescaled max-margin training objective which facilitates separation between highest scoring true and false antecedents of the current mention. A major challenge of coreference systems is resolving an anaphoric mention that has no previous head term. This paper intuitively evaluates the possibility of overcoming this challenge by means of non local decision making. It provides a conclusion that pronouns may not be the only coreferent mentions causing these errors and therefore a local model can also be tweaked with respect to a loss function to achieve this.

3 Scope and Approach

The overall task of character identification in a multiparty discourse setting, could be divided into two sub-tasks – coreference resolution and entity linking. By integrating the two modules we propose to create a system which richly annotates the dialog data, by mapping mentions to their characters introduced during the discourse.
Fig. 1. An example of character identification. All three speakers are introduced as characters before the conversation (Ross, Monica, and Joey), and two more characters are introduced during the conversation (Jack and Judy). The goal of this task is to identify each mention as one or more of these characters.

3.1 Mention Detection

To identify the mentions in a given utterance, a rule-based mention detector would be built using features such as dependency relations and named entities. As described in [1], a word sequence is considered a mention if it is a person named entity, or it is a pronoun or possessive pronoun excluding it*, or if it is in the personal noun dictionary chosen from Freebase\(^2\) and DBpedia\(^3\).

3.2 Coreference Resolution

We would be basing our coreference resolution model on the deep learning approach introduced by [4] which involves learning mention and mention pair embeddings using convolutional neural networks. These embeddings would be used to get cluster embeddings for the subsequent stage of entity linking.

3.3 Entity Linking

In the previous stage, coreference resolution groups mentions into clusters, but it does not assign character labels to the clusters, which is required for character identification. Thus, an entity linking model is required that takes the

\(^2\) https://developers.google.com/freebase/

\(^3\) http://wiki.dbpedia.org/
mention embeddings and the mention-pair embeddings generated by the CNN and classifies each mention to one of the character labels. This would involve training a feed-forward neural network with back-propagation to classify each of the mentions to an entity label.

4 Dataset

The dataset released as part of SemEval-2018, would be used for training and dev-testing the system. The validation dataset would be released in January 2018, so we’d be holding back some data to report our accuracy on. The 2018 dataset comprises first two seasons of the TV show Friends are annotated for this task. We would also be using one season of the TV show Big Bang Theory released by the shared-task organizers in 2016. Each season consists of episodes, each episode comprises scenes, and each scene is segmented into sentences.

4.1 Data Format

All datasets follow the CoNLL 2012 Shared Task data format and the following are the columns for every token in an utterance.

1. Document ID: name of the show-season ID-episode ID (e.g., friends-s01e01).
2. Scene ID: the ID of the scene within the episode.
3. Token ID: the ID of the token within the sentence.
5. Part-of-speech tag: the part-of-speech tag of the word (auto generated).
6. Constituency tag: the Penn Treebank style constituency tag (auto generated).
7. Lemma: the lemma of the word (auto generated).
8. Frameset ID: not provided.
9. Word sense: not provided.
10. Speaker: the speaker of this sentence.
11. Named entity tag: the named entity tag of the word (auto generated).
12. Entity ID: the entity ID of the mention, that is consistent across all documents.

5 Evaluation

As required by the shared task submission, we would be reporting the following metrics on the held-out dataset, using the provided evaluation script.

1. The label accuracy considering only 7 entities, that are the 6 main characters (Chandler, Joey, Monica, Phoebe, Rachel, and Ross) and all the others as one entity.
2. The macro average between the F1 scores of the 7 entities.

---

4 https:// github.com/ emorynlp/ character-mining/blob/master/md/ corpus.md
3. The label accuracy considering all entities, where characters not appearing in the training data are grouped as one entity, others.
4. The macro average between the F1 scores of all entities.
5. The F1 scores for 7 entities.
6. The F1 scores for all entities.

6 Proposed Tools and Resources

1. SpaCy \(^5\) for dependency parsing
2. Pattern \(^6\) and Stanford CoreNLP \(^7\) for parsing, tokenization, POS tags, chunking, PNP tags and lemmata.
3. Machine Learning functions and utilities in Python from : scikit-learn, keras, pyTorch, tensorflow, numpy, matplotlib etc.

References


\(^5\) https://spacy.io/
\(^6\) https://www.clips.uantwerpen.be/pages/pattern-en
\(^7\) https://stanfordnlp.github.io/CoreNLP/