Coreference Resolution Using AdapterFusion-based Multi-Task learning

Sobha Lalitha Devi, Vijay Sundar Ram R and Pattabhi RK Rao
AU-KBC Research Centre,
MIT Campus of Anna University, Chennai
{sobha, sundar, pattabhi}@au-kbc.org

Abstract

End-to-end coreference resolution is the task of identifying the mentions in a text that refer to the same real world entity and grouping them into clusters. It is crucially required for natural language understanding tasks and other high-level NLP tasks. In this paper, we present an end-to-end architecture for neural coreference resolution using AdapterFusion, a new two stage learning algorithm that leverages knowledge from multiple tasks. First task is in identifying the mentions in the text and the second to determine the coreference clusters. In the first task we learn task specific parameters called adapters that encapsulate the task-specific information and then combine the adapters in a separate knowledge composition step to identify the mentions and their clusters. We evaluated it using FIRE corpus for Malayalam and Tamil and we achieved state of art performance.

1 Introduction

Coreference resolution is the task of grouping mentions in a text that refer to the same real-world entity into clusters (Poesio et al., 2016). End-to-end coreference resolution is the task of identifying and grouping mentions in a text such that all mentions in a cluster refer to the same entity.

Many traditional coreference systems, either rule based (Haghighi and Klein, 2009; Lee et al., 2011) or learning-based (Bengtson and Roth, 2008; Fernandes et al., 2012; Durrett and Klein, 2013; Bjorkelund and Kuhn, 2014), solve the problem in two separate stages: Initially a mention identifier to identify entity mentions from the text, then a coreference resolver to cluster identified mentions. At both stages, they depend upon fine-grained, conjoined features using heuristics. This two staged pipeline approach can cause cascading errors, and since both stages has dependency over syntactic parser and handcraft features, it is difficult to scale up the system for new data sets and languages.

The first state of the art end-to-end neural coreference resolution system was proposed by Lee et al. (2017). Here all text spans are considered as potential mentions and therefore eliminate the need of carefully hand-engineered mention detection systems. In addition, the representation of pre-trained word embeddings and deep neural networks have made the model use a minimal set of hand-engineered features (speaker ID, document genre, span distance, and span width). The core of the end-to-end neural coreference resolver is the scoring function to compute the mention scores for all possible spans and the antecedent scores for a pair of spans.

One of the major challenges of coreference resolution is that most mentions in the document are singleton or non-anaphoric, i.e., not coreferent with any previous mention (Wiseman et al., 2015). Since the data set only have annotations for mention clusters, the end-to-end coreference resolution system needs to detect mentions, detect anaphoricity, and perform coreference linking. Therefore, good designs of the scoring architecture and the learning strategy for both mention detection and antecedent identification needs to be arrived at given only the gold cluster labels. To this end, we propose to use AdapterFusion-based multi task learning methodology proposed by (Pfeiffer et al, 2021) for an end-to-end coreference resolution system.

2 Our Approach

For developing an end-to-end coreference resolution system, we used AdapterFusion-based multi-task learning methodology (Pfeiffer et al, 2021). This transfer learning method has two stage learning, knowledge extraction stage and knowledge composition step. A brief description on transfer learning, Adapter and Adapterfusion are given below.

The two predominant approaches in transfer learning for sharing knowledge across multi-task are

- **Sequential Fine-tuning:** This involves sequentially updating all the weights of the model on each task. This approach works well for two sequential tasks and beyond that leads to catastrophic forgetting.

- **Multi-Task Learning (MTL):** All tasks are trained simultaneously with the aim of learning a shared representation that will enable the model to generalize better on each task. More tasks cannot be added as MTL requires to simultaneously access all the tasks.

2.1 Adapters

To overcome the limitation in Sequential Fine-tuning and MTL, Houlsby et al (2019) introduced adapters, which do not require fine-tuning of all the parameters of the pre-trained model and instead introduce a small number of task specific parameters while keeping the underlying pre-trained language model fixed. Adapters share a large set of parameters Θ across all tasks and introduce a small number of task-specific parameters Φ_n. While Θ represents the weights of a pre-trained model (e.g., a transformer), the parameters Φ_n, where $n \in \{1, \ldots, N\}$, are used to encode task-specific representations in intermediate layers of the shared model. There are two variants of adapters, namely Single task Adapter, where different Adapters are trained for each of the N task and Multiple Task Adapter, where N task is trained in parallel (Stickland and Murray, 2019).

2.2 Adapter Fusion

To maximize the transfer of knowledge across tasks, without suffering from catastrophic forgetting and difficulties in dataset balancing, AdapterFusion was introduced by Pfeiffer et al (2021). After the training of the task specific Adapters, these Adapter are combined using AdapterFusion. Once training for the adapters Φ_m and again for training Fusion parameters Ψ_m, which learn to compose the information stored in the N task adapters. It learns to compose the N task adapters Φ_n and the shared pre-trained model Θ, by introducing a new set of weights Ψ. These parameters learn to combine the adapters as a dynamic function of the target task data.

3 Experiment

End-to-end coreference resolution requires the following task, named entity recognizer, pronominal resolution, noun-noun anaphora resolution and coreference entity clustering. We train each task as a single Task Adapter. The adapters are combined in the AdapterFusion task and Fusion parameters Ψ is learned.

3.1 Experimental Setup

In training all the four adapters, we use XLM-R as the pre-trained language model. We train them with reduction factors {2, 16, 64} and learning rate 0.0001 with AdamW and a linear learning rate decay. We train for a maximum of 30 epochs with early stopping as used by (Pfeiffer et al, 2021).

For AdapterFusion, We used a learning rate of $5e{-}5$. We trained for a maximum of 10 epochs with early stopping. While we initialize Q and K randomly, we initialize V with a diagonal of ones and the rest of the matrix with random weights having a small norm ($1e-6$) as mentioned by Pfeiffer et al, (2021).

4 Dataset

The availability of coreference annotated data is the major bottleneck in Indian languages. Ontonotes, have created standard coreference annotated data and has made available for Arabic, Chinese and English.

In Indian languages, following the convention adopted in Ontonotes, data was created and shared in ICON 2011 shared task. It has anaphora annotated sentences in Bengali, Hindi and Tamil. SocAnaRes (20221 and 2022) has anaphora annotated microblog data from Twitter, in Hindi, Malayalam and Tamil (Sobha, 2022).

In the present study we have used SocAnaRes2022 corpus (FIRE 2021 and 2022 Corpus) which has conversation tweets collected by recursively retrieving the parent tweets to construct the full conversational tree structure. To construct the full tree, 5 iterations were performed and considered each chain of tweets as a document. Out of the three languages we have taken Malayalam and Tamil for our study. It has anaphors and antecedent annotated. For the present task, we have annotated named entities, noun-noun anaphors which includes definite description using PALinkA (Orasen, 2003), which is an open source tool from University of Wolverhampton.

The corpus statistics is given in the table 1, below.

Description	**Malayalam**	**Tamil**
No. of files	370	482
No. of Tweets	1880	2855

Table1: Corpus Statistics

S.No	**Type**	**Number of Occurrence**	
		Malayalam	**Tamil**
1	Noun-Noun Anaphora	1854	2424
2	Total No. of Anaphors	1170	991
3	Anaphors having antecedent	455	665

Table 2: Distribution of Coreferential Entities

5 Results and Discussion

The data was randomly divided into 80-20 (document-wise). 80% of the documents were used for training and the 20% is used for testing. Both Malayalam and Tamil (the languages under study) are morphologically rich and highly agglutinative.

We devised two types of experiments using two different data sets

- data set only with words
- morph-level where word is morphologically analysed and separated as root word and suffixes.

Since Malayalam and Tamil are morphologically and has high agglutination. We preprocessed the data by tokenizing the agglutinated words into separate words and morphologically segmenting the inflected words.

The performance measures for these two experiments for named entity recognizer (NER), pronominal resolution, noun-noun anaphora resolution both for Malayalam (MAL) and Tamil (TAM) are presented in table 3. Coreference chains are usually evaluated with metrics such as MUC, B^3 and $CEAF_e$. Coreference entity clustering is evaluated with these measures and presented in table 4.

SNo	**Exp**	**NER**						**Pronominal Resolution**						**Noun-noun anaphora resolution**					
		MAL			**TAM**			**MAL**			**TAM**			**MAL**			**TAM**		
		Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
1	AF word - level	72	61	66	77	64	70	39	28	33	41	32	36	65	51	56	69	54	60
2	AF morp h- level	78	69	73	84	71	77	42	33	36	46	42	44	69	58	63	74	63	68

Table 3: Evaluation of NER, pronominal resolution, noun-noun anaphora resolution

(AF- AdapterFusion, Pr-Precision, Re-Recall, F1- F1measure, MAL-Malayalam, TAM-Tamil)

The results in table 3 show that the system has learned the three tasks. The results obtained for the tweet data are encouraging and higher than the results obtained in the shared task. In table 4, we have the performance scores for the coreference chains.

S.No	Exp	MUC						B^3						$CEAF_e$					
		ML			TA			ML			TA			ML			TA		
		Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
1	AF Word -level	12	10	11	19	13	15	33	24	28	42	31	36	18	25	21	24	33	28
2	AF morp h- level	15	13	1	21	17	19	36	28	32	47	34	39	22	29	25	29	38	33

Table 4: Evaluation of coreference chains

From table 3 and table 4, it is very evident that the AdapterFusion learning trained with morph-level data has performed better. There are only few published results in Tamil Coreference resolutions (Vijay and Sobha, 2017). And the present results are better than the published results. Coreference resolution evaluation using MUC, B^3 and $CEAF_e$ metrics shows encouraging results. The major challenge in microblogs data is atleast 20% of coreferential entities have their antecedents occurring in the previous tweets or understood with world knowledge. This affects the resolution of anaphoricity, both for pronominal and noun – noun anaphora. The non-standard writing style, dialect variation and code mixed make this task more challenging.

Conclusion

In this paper, we present an end to end architecture for neural coreference resolution using AdapterFusion, a new two stage learning algorithm that leverages knowledge from multiple tasks. First task is in identifying the mentions in the text and the second to determine the coreference cluster. We evaluated using FIRE corpus for Malayalam and Tamil and we achieved state of art performance.

Limitation

Genitive case marker drop is a common phenomenon in Tamil and it affects the pronominal resolution module. The module fails to identify the antecedent, if the antecedent is a part of possessive NP and the genitive marker is dropped in the possessive noun.

In Malayalam, which is highly agglutinative, noun and verb conjoin together and occur as a single token. In this instance, if the antecedent is part of the token, it is not identified.

Spell variation, Anglicization are not handled and this affects the noun-noun anaphora resolution. Names and their nick names such as 'Gandhi' is popularly called as 'Mahatma', 'Baapuji'; 'Subhas Chandra bose' is popularly called as 'Netaji', 'Vallabhbhai Patel' is known as 'Iron man of India', are not handled in noun-noun anaphora resolution. The world knowledge to resolve these coreferential entities is not provided to the system. This algorithm requires large GPU resources.

References

Eric Bengtson and Dan Roth. 2008. Understanding the Value of Features for Coreference Resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 294–303, Honolulu, Hawaii. Association for Computational Linguistics

Anders Björkelund and Jonas Kuhn. 2014. Learning Structured Perceptrons for Coreference Resolution with Latent Antecedents and Non-local Features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 47–57, Baltimore, Maryland. Association for Computational Linguistics.

Greg Durrett and Dan Klein. 2013. Easy Victories and Uphill Battles in Coreference Resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1971–1982, Seattle, Washington, USA. Association for Computational Linguistics.

Eraldo R. Fernandes,;Cícero N. dos Santos and Ruy L. Milidiú. 2012. Latent ´ structure perceptron with feature induction for unrestricted coreference resolution. *In Proceedings of CoNLL '12: Joint*

Conference on EMNLP and CoNLL – Shared, Pages 41–48

Aria Haghighi and Dan Klein. 2010. 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*, pages 385–393, Los Angeles, California. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzkebski, BrunaMorrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. *In Proceedings of the 36th International Conference on Machine Learning, ICML* 2019, 9-15 June 2019, Long Beach, California, USA, pages 2790–2799

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. *In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.

ConstantinOrasan. 2003. PALinkA: a highly customizable tool for discourse annotation. *In Proceedings of the 4th SIGdial Workshop on Discourse and Dialog*, pages 39 - 43, Sapporo, Japan.

Jonas Pfeiffer, AishwaryaKamath , Andreas Ruckle , KyunghyunCh , IrynaGurevych. 2021. Adapterfusion: Non-destructive task composition for transfer learning. *In Proceedings 365 of the 16th Conference of the European Chapter of the Association for Computational Linguistics*: Main Volume (pp. 487–503).

Massimo Poesio, Roland Stuckardt, YannickVersley. (Eds.). 2016. *Anaphora Resolution: Algorithms, Resources, and Evaluation*. Springer Verlag.

SobhaLalithaDevi . 2022. Anaphora Resolution from Social Media Text in Indian Languages (SocAnaRes-IL): 2 nd Edition-Overview. *In the Forum for Information Retrieval and Evaluation-2022*, ISI, Kolkata, India.

Asa Cooper Stickland and Iain Murray. 2019. BERT and pals: Projected attention layers for efficient adaptation in multi-task learning. *In Proceedings of the 36th International Conference on Machine Learning, ICML* 2019, 9-15 June 2019, Long Beach, California, USA, pages 5986–5995.

Sam Wiseman, Alexander M Rush, Stuart Shieber, and Jason Weston. 2015. Learning anaphoricity and antecedent ranking features for coreference resolution. *In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (Volume 1: Long Papers), volume 1, pages 1416–1426.

Vijay Sundar Ram and Sobha Lalitha Devi, Coreference Resolution in Tamil Text, In *Proceedings of the 14th International Conference on Natural Language Processing* (ICON-2017), 2017, India, pp 392-401