SemEval 2018 Task 4: Character Identification on Multiparty Dialogues

Jinho D. Choi
Computer Science
Emory University
Atlanta, GA 30322
jinho.choi@emory.edu

Henry Y. Chen
Information Security
Snap Inc.
Santa Monica, CA 90405
henry.chen@snapchat.com

Abstract

Character identification is a task of entity linking that finds the global entity of each personal mention in multiparty dialogue. For this task, the first two seasons of the popular TV show *Friends* are annotated, comprising a total of 448 dialogues, 15,709 mentions, and 401 entities. The personal mentions are detected from nominals referring to certain characters in the show, and the entities are collected from the list of all characters in those two seasons of the show. This task is challenging because it requires the identification of characters that are mentioned but may not be active during the conversation. Among 90+ participants, four of them submitted their system outputs and showed strengths in different aspects about the task. Thorough analyses of the distributed datasets, system outputs, and comparative studies are also provided. To facilitate the momentum, we create an open-source project for this task and publicly release a larger and cleaner dataset, hoping to support researchers for more enhanced modeling.

1 Introduction

Most of the earlier works in natural language processing (NLP) had focused on formal writing such as newswires, whereas many recent works have targeted at colloquial writing such as text messages or social media. Since the evolution of Web 2.0, the amount of user-generated contents involving colloquial writing has exceeded the one with formal writing. NLP tasks are relatively well-explored at this point for certain types of colloquial writing i.e., microblogs and reviews (Ritter et al., 2011; Kong et al., 2014; Ranganath et al., 2016; Shin et al., 2017). However, the genre of multiparty dialogue is still under-explored, even though digital contents in dialogue forms keep increasing at a faster rate than any other types of writing.[1] This inspires us to create a new task called character identification that aims to link personal mentions (e.g, *she*, *mom*) to their global entities across multiple dialogues, where the entities indicate the specific characters referred by those mentions (e.g., *Judy*).

Due to the nature of multiparty dialogue where several speakers take turns to complete a context, character identification is a crucial step for adapting higher-end NLP tasks (e.g., summarization, question answering, machine translation) to this genre. It can also bring another level of sophistication to intelligent personal assistants or tutoring systems. This task is challenging because it needs to process through colloquialism that includes slangs, grammar mistakes, and/or rhetorical questions, as well as to handle cross-document resolution for the identification of entities that are mentioned but may not be actively participating during the conversation. Nonetheless, we believe that models produced by this task will remarkably enhance inference on dialogue contexts (e.g., business meetings, doctor-patient conversations) by providing finer-grained information about individual characters.

Section 2 illustrates the task of character identification and explains the key differences between it and other types of entity linking tasks. Section 3 describes the corpus, based on TV show transcripts, used for this task with annotation details. Section 4 gives brief overviews of the systems participated in this shared task. Section 5 explains the evaluation metrics and the results produced by those systems. Finally, Section 6 gives thorough analysis and comparative studies between these systems. This task was originally conducted at CodaLab.[2] The latest dataset and the system outputs can be found from our open source project, Emory NLP.[3]

[1] https://medium.com/hijiffy/10-graphs-that-show-the-immense-power-of-messaging-apps-4a41385b24d6

[2] https://competitions.codalab.org/competitions/17310

[3] https://github.com/emorynlp/semeval-2018-task4

Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), pages 57–64
New Orleans, Louisiana, June 5–6, 2018.

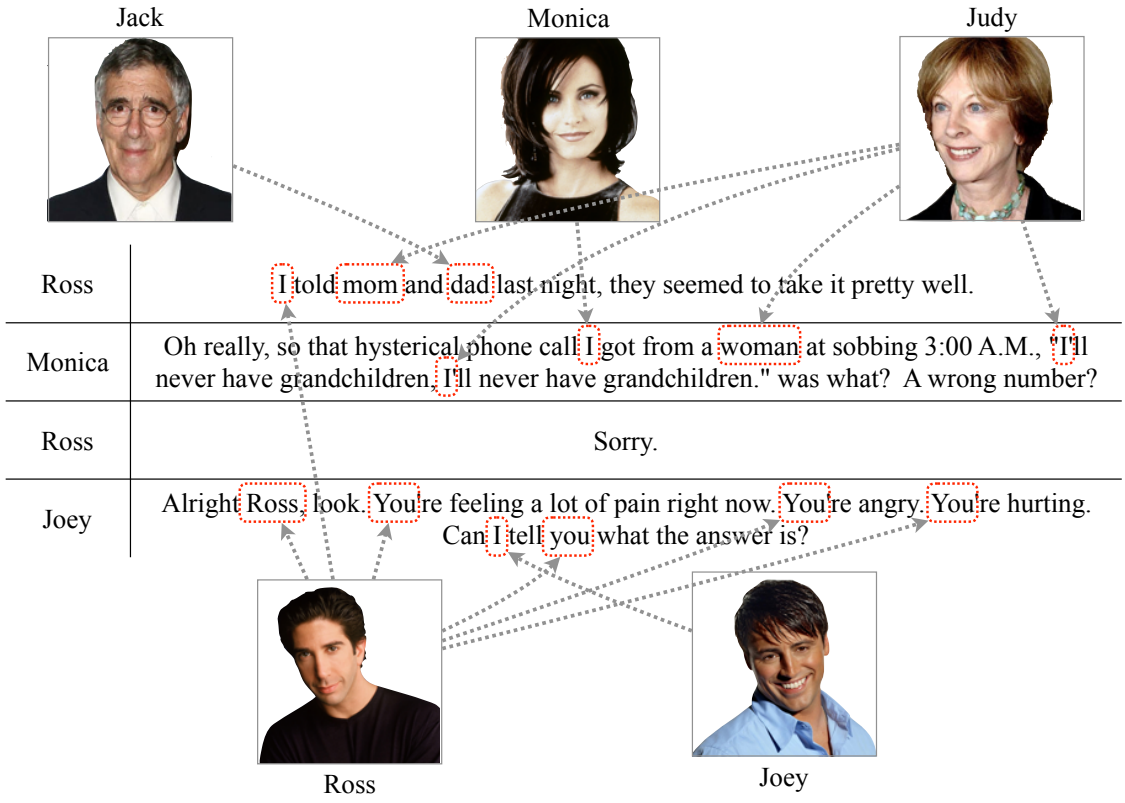

Figure 1: An example of character identification, excerpted from the Season 1 Episode 1 of *Friends*, where mentions are indicated in red boxes and entities are linked by arrows.

2 Task Description

Let a mention be a nominal that refers to a singular or a collective entity (e.g., *she*, *mom*, *Judy*), and an entity be the actual person that the mention refers to. Given a dialogue transcribed in text where all mentions are detected, the objective is to find the entity for each mention, who can be either active or passive in the dialogue. In Figure 1, entities such as *Ross*, *Monica*, and *Joey* are the active speakers of the dialogue, whereas *Jack* and *Judy* are not although they are passively mentioned as *mom* and *dad* in this context. Linking such mentions to their global entities demands inferred knowledge about the kinship from other dialogues, challenging cross-document resolution. Thus, character identification can be viewed as an entity linking task that aims for holistic understanding in multiparty dialogue.

Most of previous works on entity linking have focused on Wikification, which links named entity mentions to their relevant Wikipedia articles (Mihalcea and Csomai, 2007; Ratinov et al., 2011; Guo et al., 2013). Unlike Wikification where most entities come with structured information from knowledge bases (e.g., Infobox, Freebase, DBPedia), entities in character identification have no such precompiled information, which makes this task even more challenging. It is similar to coreference resolution in a sense that it groups mentions into entities, but distinguished because this task requires to identify each mention group as a known person. In Figure 1, coreference resolution would give a cluster of the four mentions, {*mom*, *woman*, *I*, *I*}; however, it would not identify that cluster to be the entity *Judy*, which in this case is not possible to identify without getting contexts from other dialogues.

3 Corpus

The character identification corpus was first created by collecting transcripts from the popular TV show, *Friends* (Chen and Choi, 2016). These transcripts were voluntarily provided by fans who made them publicly available.[4] Dialogues in this corpus mimic daily conversations that are more natural and various in topics than other dialogue corpora (Janin et al., 2003; Danescu-Niculescu-Mizil and Lee, 2011; Hu et al., 2013; Kim et al., 2015; Lowe et al., 2015). Although they are scripted, the interpretation of these dialogues is no easier than unscripted

[4] http://www.livesinabox.com/friends/scripts.shtml

	Episodes	Scenes	Speakers	Utterances	Sentences	Tokens
Season 1	24	229	105	4,725	8,680	66,355
Season 2	23	219	101	4,501	7,380	65,675
Total	47	448	171	9,226	16,060	132,030

Table 1: Distributions from the subset of the character identification corpus used for this shared task.

dialogues; they not only involve as much disfluency and context switching as real dialogues do, but also include more humor, sarcasm, or metaphor. Thus, models evaluated on this corpus should give a general sense about the state of character identification on multiparty dialogue.

The original transcripts collected from the fan site were formatted in HTML; we converted them into JSON so that they could be easily processed. This structured data were then manually checked for potential errors. Table 1 shows the distributions from the subset of the character identification corpus used for this shared task. The provided dataset is divided into two seasons, each season is divided into episodes, each episode is divided into scenes, each scene contains utterances, where each utterance indicates a turn of speech.

3.1 Mention Annotation

For mention annotation, a heuristic-based mention detector was developed, which utilized dependency relations (Choi and McCallum, 2013), named entity tags (Choi, 2016), and personal noun gazetteers, then automatically detected mentions for the entire corpus. In this heuristic, a noun phrase was considered a personal mention if it was either:

1. A PERSON named entity, or
2. A pronoun or a possessive pronoun excluding the pronouns *it* and *they*, or
3. One of the personal noun gazetteers that are 603 common and singular personal nouns selected from Freebase and DBPedia.

Specific mentions such as *it* and *they* were excluded because they often referred to the ambiguous entity types, *collective*, *general*, and *other* (Section 3.2). For the quality assurance, about 10% of this pseudo annotation were randomly sampled and manually evaluated, showing a precision, a recall, and the F1-score of 97.58%, 94.34%, and 95.93%, respectively. Finally, the annotation was manually checked again while it was systematically corrected for routinely produced errors. Although mention detection was the foundational step, including it as a part of this shared task could over-complicate the evaluation. Thus, gold mentions were provided for this year's shared task such that participants could purely concentrate on the task of entity linking.

3.2 Entity Annotation

All mentions were double-annotated with their referent entities, and adjudicated upon disagreements. Annotation and adjudication tasks were conducted on Amazon Mechanical Turk. Each mention was annotated with either a primary character, that are *Ross*, *Chandler*, *Joey*, *Rachel*, *Monica*, and *Pheobe*, a secondary character (other frequently recurring characters across the show), or one of the following ambiguous types suggested by Chen et al. (2017):

- *Generic*: indicates actual characters in the show whose identities are unknown (e.g., That *waitress* is really cute, I am going to ask *her* out). Generic entities are annotated with their group names and optional numberings (e.g., `Man 1`, `Woman 1`).
- *Collective*: indicates the plural use of the pronoun *you*, which cannot be deterministically distinguished from the singular use.
- *General*: indicates mentions used in reference to a general case rather than an specific entity (e.g., The ideal *guy* you look for doesn't exist).
- *Other*: indicates all the other kinds of entities.

For this year's shared task, mentions annotated with the last three ambiguous types, *collective*, *general*, and *other*, were excluded from the dataset to reduce the high complexity of this task (Table 2).

	Primary	Secondary	Generic	Total
Season 1	5,160	2,526	178	7,864
Season 2	5,385	2,340	120	7,845
Total	10,545	4,866	298	15,709

Table 2: Distributions of the annotated entity types used for this shared task.

Speaker	Utterance
Joey	Yeah, right! ... *You*$_1$ serious?
Rachel	Everything *you*$_2$ need to know is in that first kiss.
Chandler	Yeah. For *us*$_3$, it's like the stand-up *comedian*$_4$ *you*$_5$ have to sit through before the main *dude*$_6$ starts.
Ross	It's not that *we*$_7$ don't like the *comedian*$_8$, it's that ... that's not why *we*$_9$ bought the ticket.

$\{You_1\} \rightarrow Rachel$, $\{us_3, we_{7,9}\} \rightarrow Collective$, $\{you_{2,5}\} \rightarrow General$, $\{comedian_{4,8}\} \rightarrow Generic$, $\{dude_6\} \rightarrow Other$

Table 3: Examples of the entity annotation described in Section 3.2.

	Episodes	Scenes	Entities	Mentions	Clusters$_E$	Clusters$_S$	Singleton$_E$	Singleton$_S$
Training	47	374	372	13,280	893	2,051	209	472
Evaluation	7	74	106	2,429	304	370	54	83
Total	47	448	401	15,709	1,197	2,421	263	555

Table 4: Distributions of the training and the evaluation sets in Section 3.3.

Table 3 shows examples of these ambiguous types. About 83% were assigned to the primary and secondary characters, 1.4% were assigned to *generic*, and the rest were assigned to the other ambiguous types, *collective*, *general*, and *other*. To evaluate the annotation quality, the annotation agreement scores as well as Cohen's kappa scores were measured, showing 82.83% and 79.96%, respectively.

3.3 Data Split

The corpus was split into training and evaluation sets for this shared task (Table 4). No dedicated development set was provided; participants were encouraged to use sub-parts of the training set to create their own development sets or perform cross-validation for the optimization of statistical models. Two types of datasets are provided for both training and evaluation sets, one treating each episode as an individual dialogue and the other treating each scene as an independent dialogue.[5]

Processing a larger dialogue makes coreference resolution harder because it needs to link referential mentions that are farther apart; on the other hand, each cluster comprises a greater number of mentions which can help identifying the global entity of that cluster. The numbers of clusters grouped in each dataset are shown as Clusters$_E$ and Clusters$_S$, implying episode-level and scene-level clusters, respectively. Our corpus includes singleton mentions, which take about 22% of all mentions.

3.4 Data Format

To help participants adapting their existing coreference resolution systems to this task, the original dataset in JSON was converted into the CoNLL'12 shared task format (Pradhan et al., 2012), where each column is delimited by white spaces and represents the following:

1. Season and episode ID.
2. Document ID.
3. Token ID.
4. Word form.
5. Part-of-speech tag (auto-generated).
6. Phrase structure tag (auto-generated).
7. Lemma (auto-generated).
8. Predicate sense (not provided).
9. Word sense (not provided).
10. Speaker.
11. Named entity tag (auto-generated).
12. Entity ID.

The part-of-speech tags, lemmas, and named entity tags were automatically generated by NLP4J,[6] and the phrase structure tags were produced by the Stanford parser.[7] Table 5 shows the example of the first utterance in Figure 1 in the CoNLL'12 format.

4 System Description

This section describes the top-2 scoring systems of this shared task. The AMORE-UPF is a group of researchers from the Universitat Pompeu Fabra in Spain (Section 4.1). The KNU CI is a group of researchers from Kangwon National University in South Korea (Section 4.2).

[5] Each episode consists of about 10 scenes on average.

[6] https://emorynlp.github.io/nlp4j

[7] https://nlp.stanford.edu/software/lex-parser.shtml

s1e1u38	0	0	I	PRP	(TOP(S(S(NP*)	I	-	-	Ross	*	(7)
s1e1u38	0	1	told	VBD	(VP*	tell	-	-	Ross	*	-
s1e1u38	0	2	mom	NN	(NP*	mom	-	-	Ross	*	(9)
s1e1u38	0	3	and	CC	*	and	-	-	Ross	*	-
s1e1u38	0	4	dad	NN	*)	dad	-	-	Ross	*	(10)
s1e1u38	0	5	last	JJ	(NP-TMP*	last	-	-	Ross	(TIME*	-
s1e1u38	0	6	night	NN	*)))	night	-	-	Ross	*)	-
s1e1u38	0	7	,	,	*	,	-	-	Ross	*	-
s1e1u38	0	8	they	PRP	(NP*)	they	-	-	Ross	*	-
s1e1u38	0	9	seemed	VBD	(VP*	seem	-	-	Ross	*	-
s1e1u38	0	10	to	TO	(S(VP*	to	-	-	Ross	*	-
s1e1u38	0	11	take	VB	(VP*	take	-	-	Ross	*	-
s1e1u38	0	12	it	PRP	(NP*)	it	-	-	Ross	*	-
s1e1u38	0	13	pretty	RB	(ADVP*	pretty	-	-	Ross	*	-
s1e1u38	0	14	well	RB	*)))))	well	-	-	Ross	*	-
s1e1u38	0	15	.	.	*))	.	-	-	Ross	*	-

Table 5: Example of the first utterance in Figure 1 annotated in the CoNLL'12 format.

4.1 AMORE-UPF System

The AMORE-UPF system approaches this task as a multi-class classification. It uses a bidirectional Long Short-Term Memory (LSTM) that processes the input dialogue and resolves mentions, by means of a comparison between the LSTM's hidden state, for each mention, to vectors in an entity library. In this model, learned representations of each entity are stored in the entity library, that is a matrix where each row represents an entity and whose values are learned during training (Figure 2).

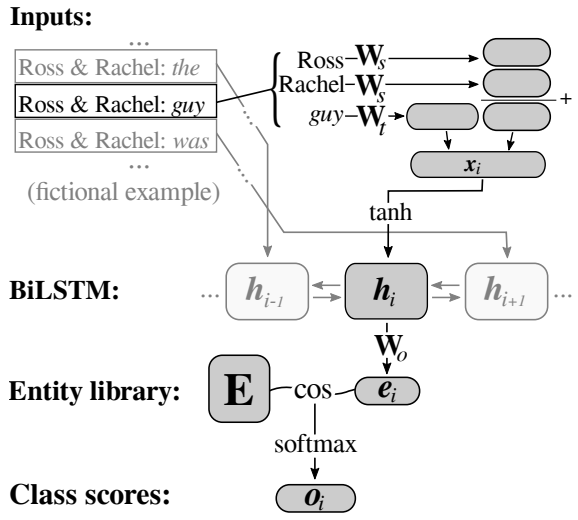

Figure 2: The overview of AMORE-UPF system.

4.2 KNU-CI System

The KNU-CI system tackles this task as a sequence-labeling problem. It uses an attention-based recurrent neural network (RNN) encoder-decoder model. The input dialogue of character identification consists of several conversations, resulting a long sequence of text. The RNN encoder-decoder model suffers from poor performance when the length of the input sequence is long. To overcome this issue, this system applies an attention, position encoding, and the self-matching network to the original RNN encoder-decoder model. As a result, the best performance is achieved by the attention-based RNN depicted in Figure 3.

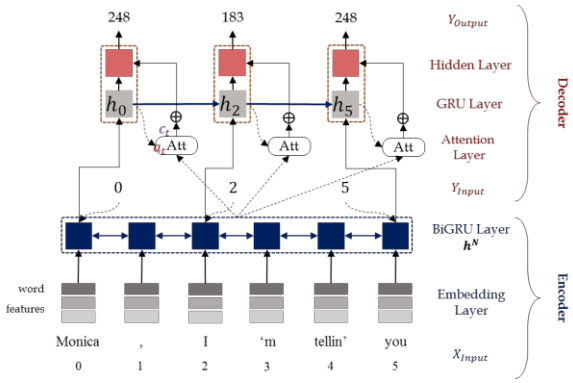

Figure 3: The overview of KNU-CI system.

5 Evaluation

Following Chen et al. (2017), the labeling accuracy (Acc) and the macro-average F1 score (F1) are used for the evaluation (C: the total number of characters, $F1_i$: the F1-score for the i'th character):

$$Acc = \frac{\text{\# of corrected identified mentions}}{\text{\# of all mentions}}$$

$$F1 = \frac{1}{C}\sum_{i=1}^{C} F1_i$$

Table 6 shows the overall scores from all submitted systems. Two types of evaluation are performed for this task. The first one is based on seven characters where six of them compose the primary characters

(Section 3.2) and every other character is grouped as one entity called `Others` (Main + Others). The other is based on 78 characters comprising all characters appeared in the dataset, except for the ones appear either in the training or the evaluation set but not both, which is grouped to the `Others` (ALL).

System	Main + Others		ALL	
	Acc	F1	Acc	F1
AMORE-UPF	77.23	79.36	**74.72**	**41.05**
KNU-CI	**85.10**	**86.00**	69.49	16.98
Kampfpudding	73.36	73.51	59.45	37.37
Zuma-AR	46.85	44.68	33.06	16.09

Table 6: Overall scores from the submitted systems.

Table 7 shows the F1 scores for the primary characters and `Others`, illustrating detailed evaluation for **Main + Others**. Table 8 gives detailed evaluation for **ALL**, showing the F1-scores for the top-12 most frequently appeared secondary characters and `Others` that appear only in the training or the evaluation set but not both. The 18 characters in these two tables comprise about 85% of all mentions.

6 Analysis

Based on the evaluation results, several interesting observations can be made for how different system architectures affect model performance on this task. The analysis in this section primarily focuses on the top-2 scoring systems, AMORE-UPF an KNU-CI, as their results vastly outperform the other two and the authors of those systems provide more detailed descriptions to the organizers.

6.1 Overall Performance

It is worth pointing out the significance of the two evaluation metrics proposed in Section 5 in terms of the model performance. The labeling accuracy indicates the raw predicative power of the model. This metric is biased towards more frequently appearing characters such as the primary characters, a total of which compose 70+% of the evaluation set. Thus, it is possible to achieve a relatively high labeling accuracy score without handling referents for the secondary characters well. On the contrary, the macro-average F1 score neutralizes the imbalance between frequently and not so frequently appearing characters. It reveals the model performance on a per-entity basis, which tends to favor transient and extra characters more because every character is treated equally in this metric.

For the overall performance, KNU-CI outperforms for **Main + Others** with the labeling accuracy of 85.10% and the macro-average F1 score of 86.00%, whereas AMORE-UPF outperforms for **ALL** with the labeling accuracy of 74.72% and the macro-average F1 of 41.05% (Table 6). All systems produce better results for **Main + Others** than **ALL**, which is expected due to the fewer number of entities to classify (7 vs 78). It is possible that KNU-CI's attention model is highly optimized for the identification of the primary characters, whereas AMORE-UPF's LSTM model distributes weights for the secondary characters more evenly, but more detailed analysis needs to be made to see the comparative strengths between these two systems.

6.2 Main + Other Identification

Table 7 depicts the strength of the KNU-CI system for the primary characters in comparisons to the others, which is attributed to its unique sequence labeling architecture and the attention mechanism. Their encoder-decoder architecture helps consolidating sequential information of the input dialogue along with the mentions. The hidden units in RNNs enable the network to aggregate character-related information and to disambiguate timeline shifts across utterances. The encoder takes the input dialogue and provides the decoder with context-rich features. Coupled with the attention mechanism, this model focuses on the primary characters; thus, it results better performance on **Main + Others**. However, this architecture is not as well-adaptive as the number of characters increases for the identification, which can be observed from the system's low macro-average F1 score for **All**.

6.3 All Character Identification

Table 8 describes the strength of the AMORE-UPF system for the secondary characters using the bidirectional LSTM model, leading it to outperform all the others for **All**. Although both AMORE-UPF and KNU-CI utilize variations of RNNs as their underlying architectures, the performance downfall is not as prominent for AMORE-UPF as the number of characters increases, thanks to its entity library. The entity library is consumed and updated as necessary given the mention embeddings. It is used to regularize training each individual character, which helps avoiding the bias towards frequently appearing characters. As the result, AMORE-UPF yields better performance for **All** while accomplishing reasonable results for **Main + Others** as well.

Character	Ross	Rachel	Chandler	Joey	Phoebe	Monica	Others
Evaluation	18.98	13.96	9.80	9.51	9.02	8.97	29.77
Training	13.93	12.37	11.43	9.43	8.79	10.61	33.44
AMORE-UPF	78.57	82.98	81.36	**79.83**	86.52	85.22	61.02
KNU-CI	**85.86**	**92.49**	**84.94**	79.67	**88.09**	**91.16**	**79.79**
Kampfpudding	73.48	70.67	79.25	63.38	79.79	73.35	74.61
Zuma-AR	38.72	43.05	43.04	36.10	42.90	46.43	51.78

Table 7: Detailed evaluation for **Main + Others** in Table 6. The Evaluation and Training rows show the percentages of individual characters appeared in the evaluation and the training set, respectively.

Character	Be	Ca	Ed	Pa	Ju	MB	Ri	Sc	Ca	Fr	Ja	OT
Evaluation	3.46	1.73	1.56	1.44	1.32	0.86	0.86	0.78	0.74	0.70	0.62	2.92
Training	1.41	1.46	1.06	0.71	1.15	0.60	1.83	0.21	0.13	0.51	0.43	13.51
AMORE-UPF	**50.00**	57.14	**80.60**	**35.56**	**72.73**	**64.52**	**80.85**	10.00	**61.54**	0.00	**42.11**	**7.89**
KNU-CI	38.46	**62.79**	73.02	15.38	42.55	0.00	66.67	**38.46**	0.00	**18.18**	16.00	0.00
Kampfpudding	31.86	33.33	68.85	33.33	60.32	50.00	61.22	10.00	0.00	0.00	23.53	0.00
Zuma-AR	0.00	12.24	44.44	0.00	27.91	15.38	77.78	0.00	38.46	0.00	12.50	0.00

Table 8: Detailed evaluation for **ALL** in Table 6. Be: Ben, Ca: Carol, Ed: Eddie, Pa: Paolo, Ju: Julie: MB: Mrs. Bing, Ri: Richard, Sc: Scott, Ca: Carl, Fr: Frank, Ja: Janice, OT: Others.

7 Conclusion

In this shared task, we propose a novel entity linking task called character identification that aims to find the global entities for all personal mentions, representing individual characters in the contexts of multiparty dialogue. Among 90+ participants signed up for this task at CodaLab, only four submitted their system outputs, which is unfortunate. However, the top-2 scoring systems depict unique strengths, allowing us to make a good analysis for this task. It would be interesting to see if the sequence labeling architecture from KNU-CI coupled with the entity library from AMORE-UPF could produce an even higher performing model for both the **Main + Other** and **All** evaluation.

To facilitate the momentum, we create an open-source project that will continuously support this task.[8] It is worth mentioning that *Character Identification* is a part of a bigger project called *Character Mining* that strives for machine comprehension on dialog.[9] Currently, this project provides more and cleaner annotation for character identification than the corpus described in Section 3, hoping to engage more researchers to this task.

[8] https://github.com/emorynlp/character-identification

[9] https://github.com/emorynlp/character-mining

References

Henry Yu-Hsin Chen and Jinho D. Choi. 2016. Character Identification on Multiparty Conversation: Identifying Mentions of Characters in TV Shows. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. SIGDIAL'16, pages 90–100.

Henry Yu-Hsin Chen, Ethan Zhou, and Jinho D. Choi. 2017. Robust Coreference Resolution and Entity Linking on Dialogues: Character Identification on TV Show Transcripts. In *Proceedings of the 21st Conference on Computational Natural Language Learning*. Vancouver, Canada, CoNLL'17, pages 216–225. http://www.conll.org/2017.

Jinho D. Choi. 2016. Dynamic Feature Induction: The Last Gist to the State-of-the-Art. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. NAACL'16.

Jinho D. Choi and Andrew McCallum. 2013. Transition-based Dependency Parsing with Selectional Branching. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. ACL'13, pages 1052–1062.

Cristian Danescu-Niculescu-Mizil and Lillian Lee. 2011. Chameleons in Imagined Conversations: A New Approach to Understanding Coordination of Linguistic Style in Dialogs. In *Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics*. CMCL'11, pages 76–87.

Stephen Guo, Ming-Wei Chang, and Emre Kiciman. 2013. To Link or Not to Link? A Study on End-to-End Tweet Entity Linking. In *Proceedings of the Conference of the North American Chapter of the*

Association for Computational Linguistics on Human Language Technology. NAACL, pages 1020–1030.

Zhichao Hu, Elahe Rahimtoroghi, Larissa Munishkina, Reid Swanson, and Marilyn A. Walker. 2013. Unsupervised Induction of Contingent Event Pairs from Film Scenes. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. EMNLP'13, pages 369–379.

Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters. 2003. The ICSI Meeting Corpus. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*. ICASSP'03, pages 364–367.

Seokhwan Kim, Luis Fernando ĎHaro, Rafael E. Banchs, Jason D. Williams, and Matthew Henderson. 2015. The Fourth Dialog State Tracking Challenge. In *Proceedings of the 4th Dialog State Tracking Challenge*. DSTC4.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A. Smith. 2014. A Dependency Parser for Tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. EMNLP, pages 1001–1012.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. SIGDIAL'15, pages 285–294.

Rada Mihalcea and Andras Csomai. 2007. Wikify!: Linking Documents to Encyclopedic Knowledge. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*. CIKM'07, pages 233–242.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. In *Proceedings of the Sixteenth Conference on Computational Natural Language Learning: Shared Task*. CoNLL'12, pages 1–40.

Suhas Ranganath, Xia Hu, Jiliang Tang, Suhang Wang, and Huan Liu. 2016. Identifying Rhetorical Questions in Social Media. In *Proceedings of the 10th International Conference on Web and Social Media*. pages 667–670.

Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and Global Algorithms for Disambiguation to Wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. ACL'11, pages 1375–1384.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named Entity Recognition in Tweets: An Experimental Study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. EMNLP, pages 1524–1534.

Bonggun Shin, Timothy Lee, and Jinho D. Choi. 2017. Lexicon Integrated CNN Models with Attention for Sentiment Analysis. In *Proceedings of the EMNLP Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. Copenhagen, Denmark, WASSA'17, pages 149–158. http://optima.jrc.it/wassa2017/.