Learning Interactions and Relationships between Movie Characters
SUPPLEMENTARY MATERIAL

Anna Kukleva[1,2] Makarand Tapaswi[1] Ivan Laptev[1]
[1]Inria Paris, France [2]Max-Planck-Institute for Informatics, Saarbrücken, Germany
akukleva@mpi-inf.mpg.de, {makarand.tapaswi,ivan.laptev}@inria.fr

We provide additional analysis of our task and models including confusion matrices, prediction examples for all our models, skewed distribution of number of samples for our classes, and diagrams depicting how we grouped the interaction and relationship classes.

A. Impact of Modalities

We analyze the impact of modalities by presenting qualitative examples where using multiple modalities help predict the correct interactions. Qualitative results presented here, refer to the quantitative performance indicated in Table 1 of the main paper. Fig. 1 shows that using dialog can help to improve predictions, Fig. 2 demonstrates the necessity of visual clip information and highlights that the two modalities are complementary. Finally, Fig. 3 shows that focusing on tracks (visual representations in which the two characters appear) provides further improvements to our model. Furthermore, Fig. 4 shows top-5 interaction classes that benefit most from using additional modalities.

Analyzing modalities. We also analyze the two models trained on only visual or only dialog cues (first two rows of Table 1). Some interactions can be recognized only with **visual** (v) features: *rides* 63% (v) / 0% (d), *walks* 29% (v) / 0% (d), *runs* 26% (v) / 0% (d); while others only with **dialog** (d) cues: *apologizes* 0% (v) / 66% (d), *compliments* 0% (v) / 26% (d), *agrees* 0% (v) / 25% (d).

Interactions that achieve non-zero accuracy with both modalities are: *hits* 64% (v) / 5% (d), *greets* 12% (v) / 57% (d), *explains* 25% (v) / 51% (d).

Additionally, the top-5 predicted classes for **visual** cues are *asks* 77%, *hits* 64%, *rides* 63%, *watches* 49%, *talks on phone* 41%; and **dialog** cues are *asks* 75%, *apologizes* 66%, *greets* 57%, *explains* 51%, *watches* 30%. As *asks* is the most common class, and *watches* is the second most common, these interactions work well with both modalities.

B. Joint Interaction and Relationships

Confusion matrices. Fig. 5 shows the confusion matrix in the top-15 most commonly occurring interactions on the validation and test sets. We see that multiple dialog based interactions (*e.g.* talks to, informs, and explains) are often confused. We also present confusion matrices for relationships in Fig. 6. A large part of the confusion is due to lack of sufficient data to model the tail of relationship classes.

Qualitative examples. Related to Table 2 of the main paper, Fig. 7 shows some examples where interaction predictions improve by jointly learning to model both interactions and relationships. Similarly, Fig. 8 shows how relationship classification benefits from our multi-task training setup.

C. Examples for *Who* is Interacting

Empirical evaluation shows that the knowledge about the relationship is important for localizing the pair of characters (Table 6 of the main paper). In Fig. 9, we illustrate an example where the dad walks into a room, sees his daughter *with* someone, and asks questions (see figure caption for details).

Finally, in Fig. 10, we show an example where the model is able to correctly predict all components (interaction class, relationship type and the pair of tracks) in a complex situation with more than 2 people appearing in the clip.

D. Dataset Analysis

Fig. 11 and Fig. 12 show normalized distributions for the number of samples in each class for train, validation and test sets of interactions and relationships respectively. As can be seen the most common classes appear many more times than the others. Data from a complete movie belongs to one of the three train/val/test sets to avoid model bias on the plot and characters behaviour. Notably, this means that the relative ratios between number of samples per class are also not necessarily consistent making the dataset and task even more challenging.

In the main paper, we described our approach to group over 300 interactions into 101 classes, and over 100 relationships into 15. We use radial tree diagrams to depict the groupings for interaction and relationship labels, visualized in Fig. 13 and 14 respectively.

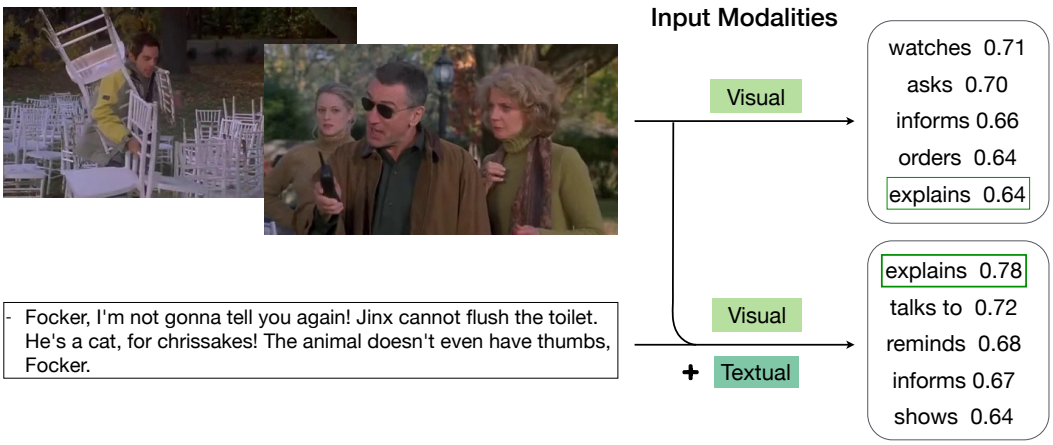

Figure 1: Improvement in prediction of **interactions** by including textual modality in addition to visual. The model learns to recognize subtle differences between interactions based on dialog. The example is from *Meet the Parents (2000)*.

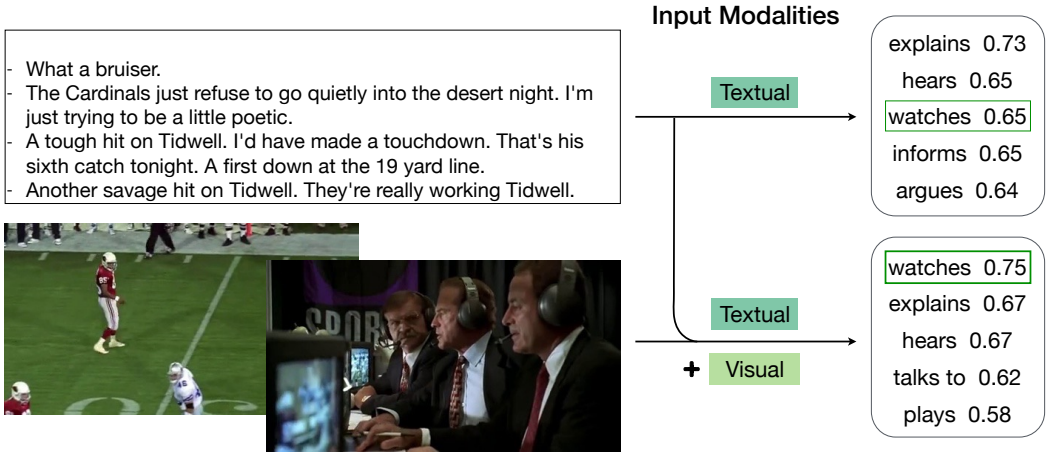

Figure 2: Improvement in prediction of **interactions** by including visual modality in addition to textual. The top-5 predicted interactions reflect the impact of visual input rather than relying only on the dialog. The example is from *Jerry Maguire (1996)*.

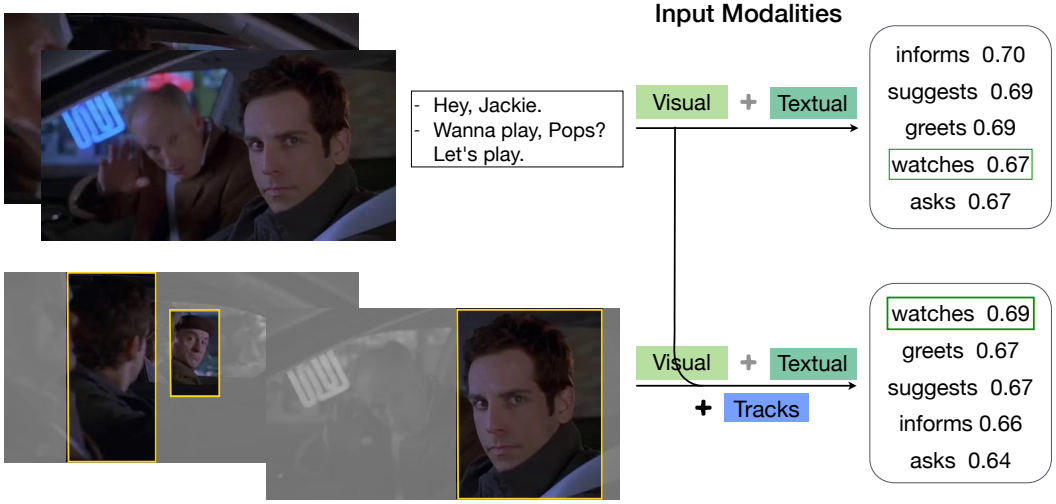

Figure 3: Improvement in prediction of **interactions** by including the pair of tracks modality in addition to visual and textual cues. The model can concentrate its attention on visual cues for the two people of interest instead of looking only at the clip level. The example is from *Meet the Parents (2000)*.

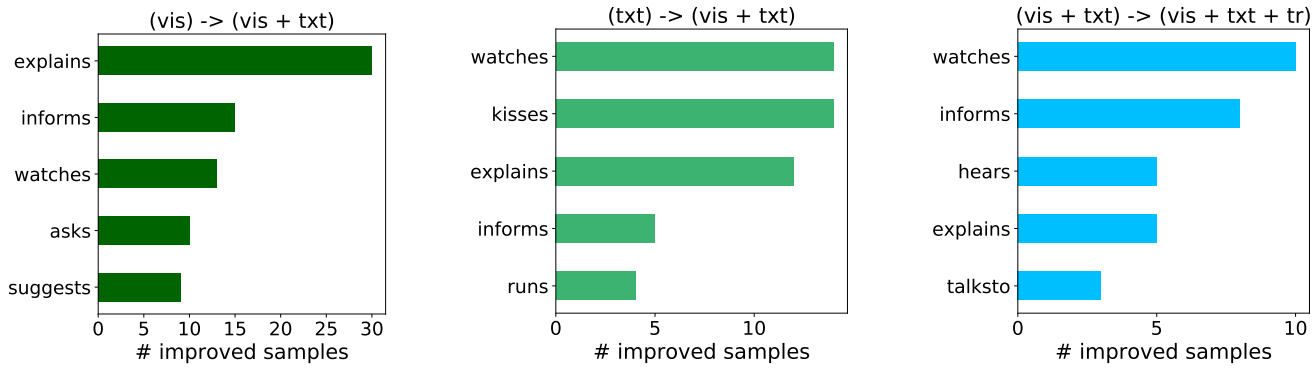

Figure 4: Each plot shows 5 **interaction** classes that have the most number of improved instances by including an additional modality. Specifically, the x-axis denotes the number of samples in which interaction prediction performance improves. **Left**: From only *visual* clip representation to *visual and textual*. As expected, using dialogues in addition to video frames boosts performance for classes that rely on dialog *e.g. explains, informs*. **Middle**: From only *textual* clip representation to *visual and textual*. Visual clip representations influence classes as *kisses, runs* during which people usually do not talk (dialog modality filled with zeros). **Right**: Finally, including all three modalities *visual, textual, tracks* improves performance over using *visual and textual*. Track pair localization improves recognition of classes typically used in group activities.

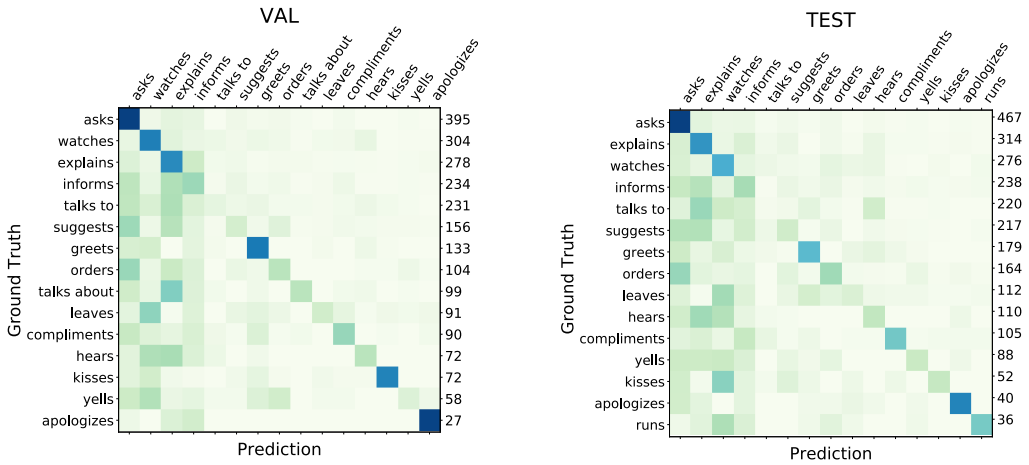

Figure 5: Confusion matrices for top-15 most common interactions for validation set (left) and test set (right). Model corresponds to the "Int. only" performance of 26.1% shown in Table 2. Numbers on the right axis indicate number of samples for each class.

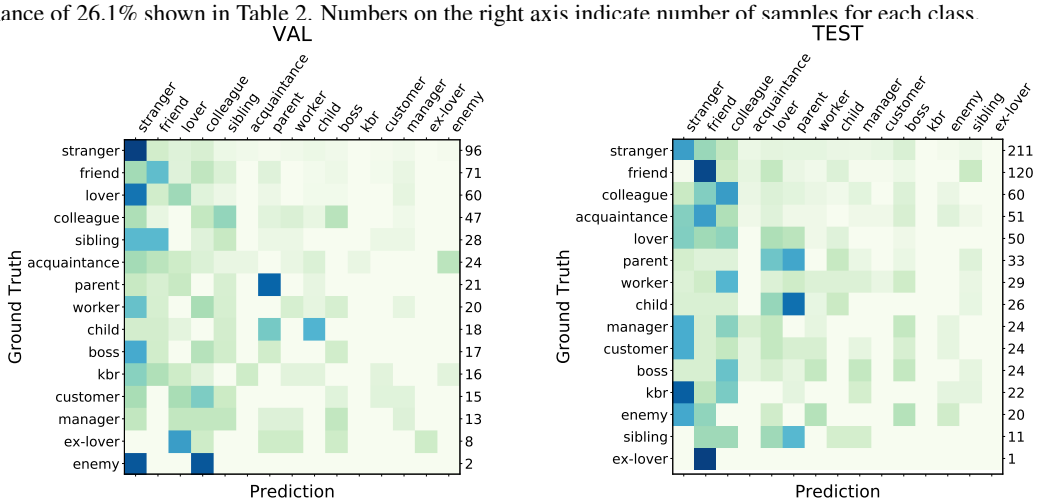

Figure 6: Confusion matrices for all relationships for validation set (left) and test set (right). Model corresponds to the "Rel. only" performance of 26.8% shown in Table 2. Numbers on the right axis indicate number of samples for each class.

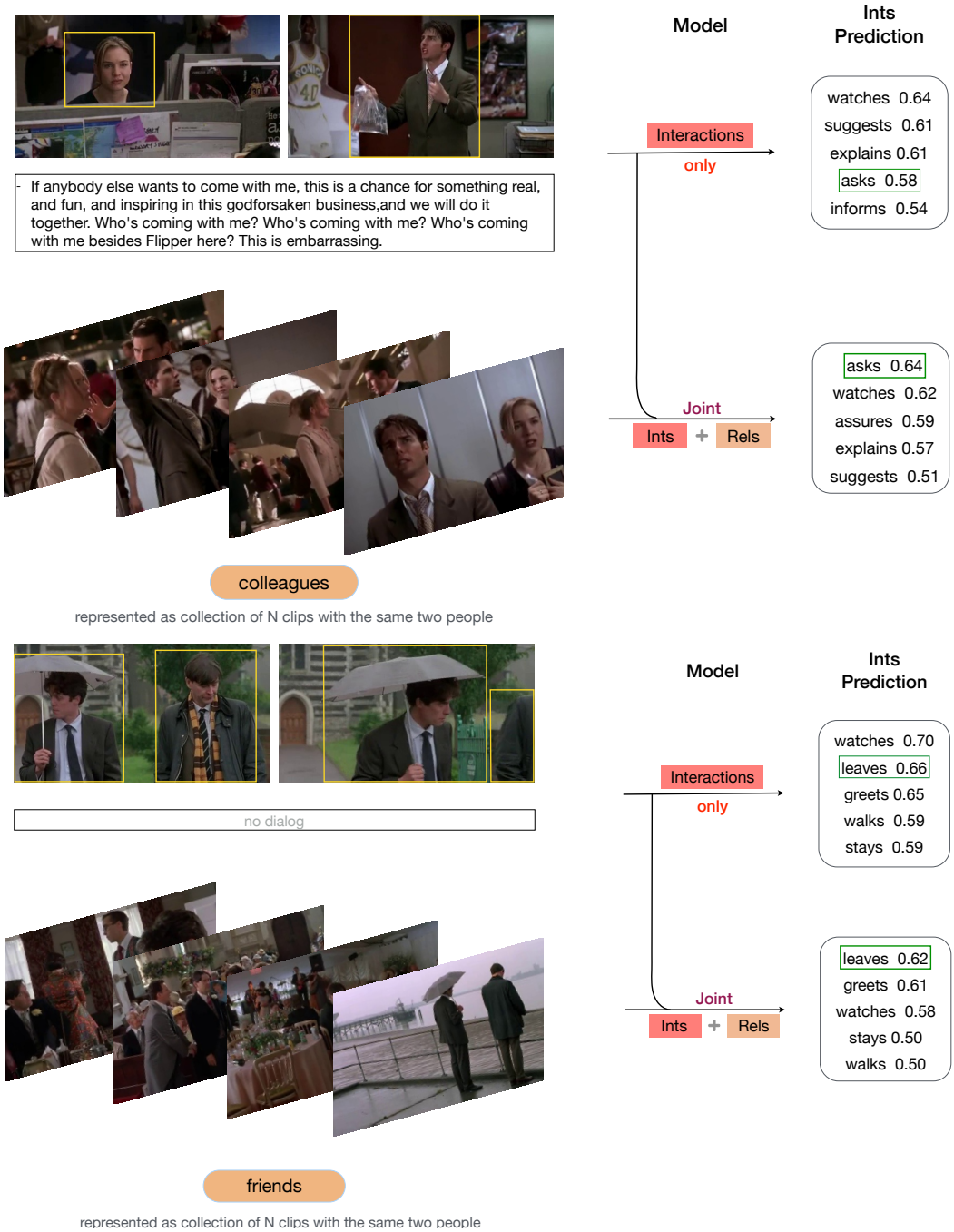

Figure 7: We show examples where training to predict interactions and relationships jointly helps improve the performance of **interactions**. **Top**: In the example from *Jerry Maguire (1996)*, the joint model looks at several clips between Dorothy and Jerry and is able to reason about them being *colleagues*. This in turn helps refine the interaction prediction to *asks*. **Bottom:** In the example from *Four Weddings and Funeral (1994)*, the model observes several clips from the entire movie where Charles and Tom are friends, and reasons that the interaction should be *leave* (which contains the *leave together* class). Note that there is no dialog for this clip.

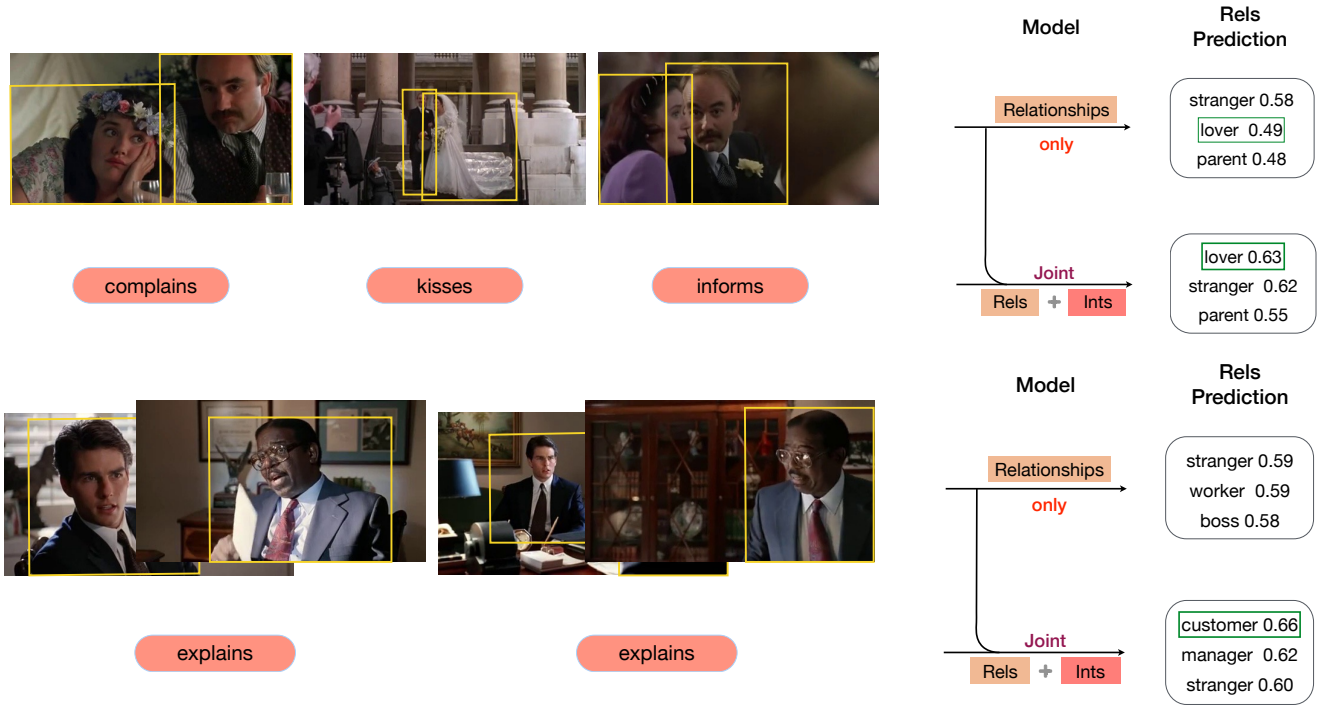

Figure 8: We show examples where training to predict interactions and relationships jointly helps improve the performance of **relationships**. **Top**: In the movie *Four Weddings and Funeral (1994)*, clips between Bernard and Lydia exhibit a variety of interactions (*e.g.* kisses) that are more typical between *lovers* than *strangers*. **Bottom**: In the movie *The Firm (1993)*, Frank and Mitch meet only once for a consultation, and are involved in two clips with the same interaction label *explains*. Our model is able to reason about this interaction, and it encourages the relationship to be *customer* and *manager*, instead of *stranger*.

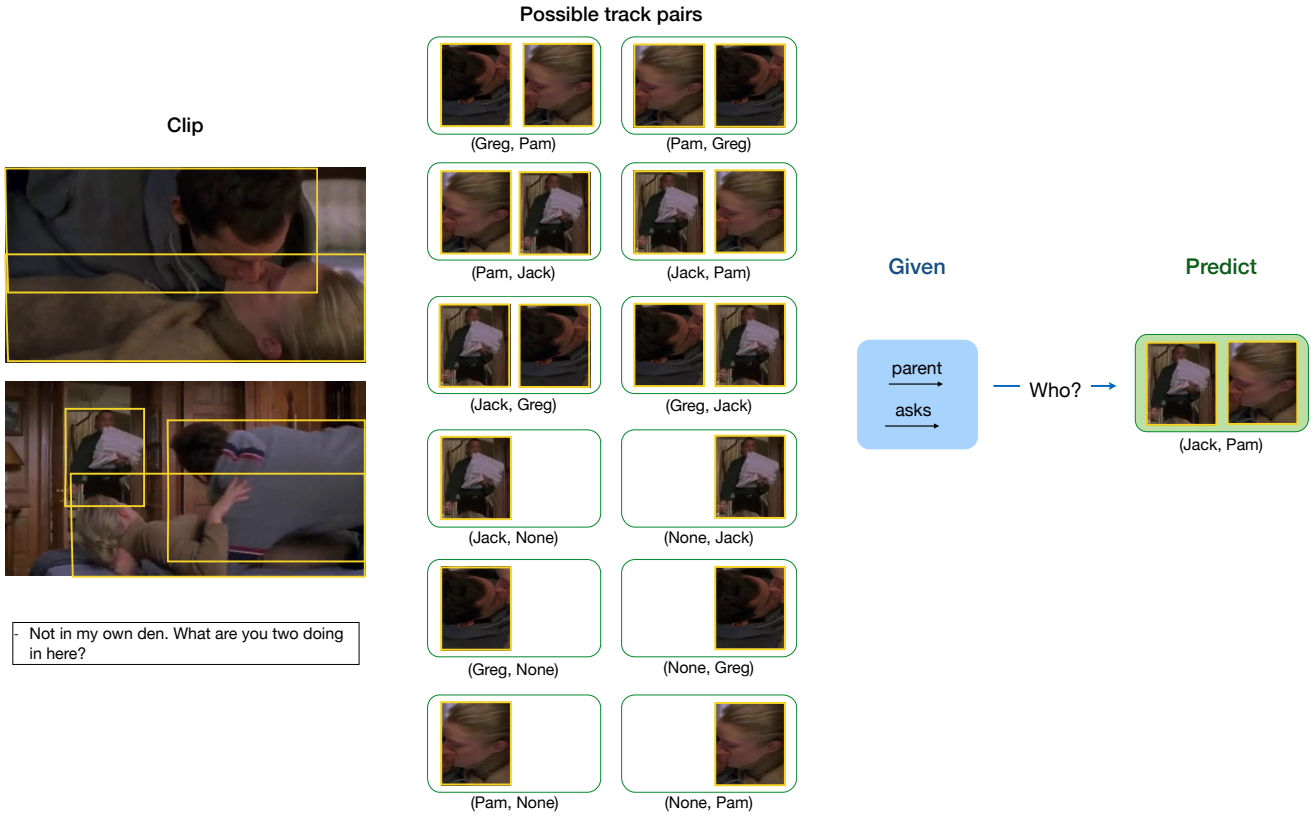

Figure 9: We illustrate an example from the movie *Meet the Parents (2000)* where a father (Jack) walks into a room while his daughter (Pam) and the guy (Greg) are kissing. Our goal is to predict the two characters when the interaction and relationship labels are provided. In this particular example, we see that Dad asks Pam a question (What are you two doing in here?). Note that their relationship is encoded as (Pam $\rightarrow$ child $\rightarrow$ Jack), or equivalently, (Jack $\rightarrow$ parent $\rightarrow$ Pam). When searching for the pair of characters with a given interaction **asks** and relationship as **parent**, our model is able to focus on the question at the clip level as it is asked by Jack in the interaction, and correctly predict *(Jack, Pam)* as the ordered character pair. Note that our model not only considers all possible directed track pairs (*e.g.* (Greg, Pam) and (Pam, Greg)) between characters, but also singleton tracks (*e.g.* (Jack, None)) to deal with situations when a person is absent due to failure in tracking or does not appear in the scene.

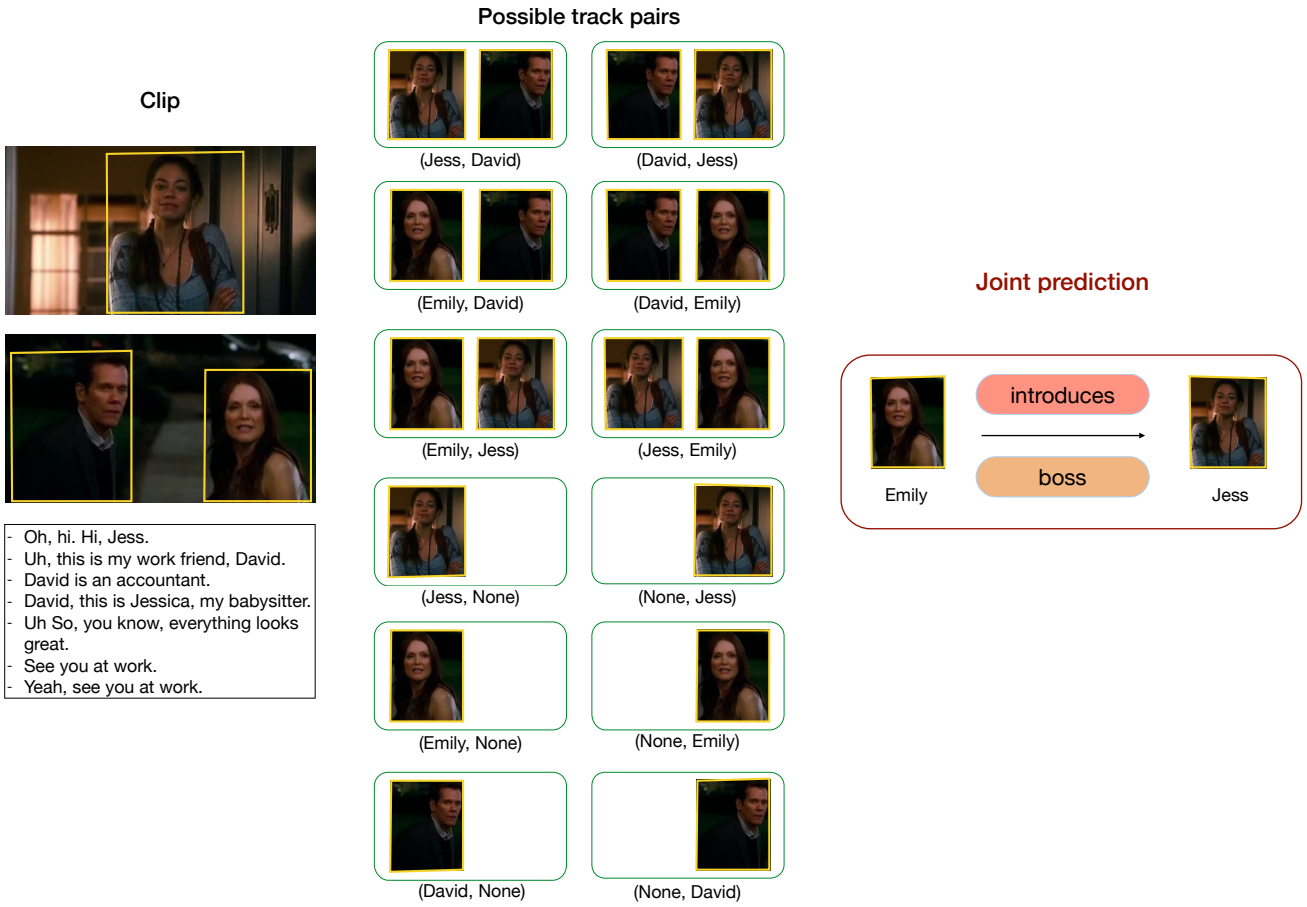

Figure 10: We present an example where our model is able to correctly and jointly predict all three components: track pair, interaction class and relationship type for the clip obtained from the movie *Crazy, Stupid, Love (2011)*. This clip contains three characters which leads to 12 possible track pairs (including singletons to deal with situations when a person is absent due to failure in tracking or does not appear in the scene). The model is able to correctly predict the two characters, their order, interaction and relationship. In this case, *Emily* **introduces** David **to** *Jess*. Jess is also her hired babysitter, and thus their relationship is – *Emily* is **boss** of *Jess*.

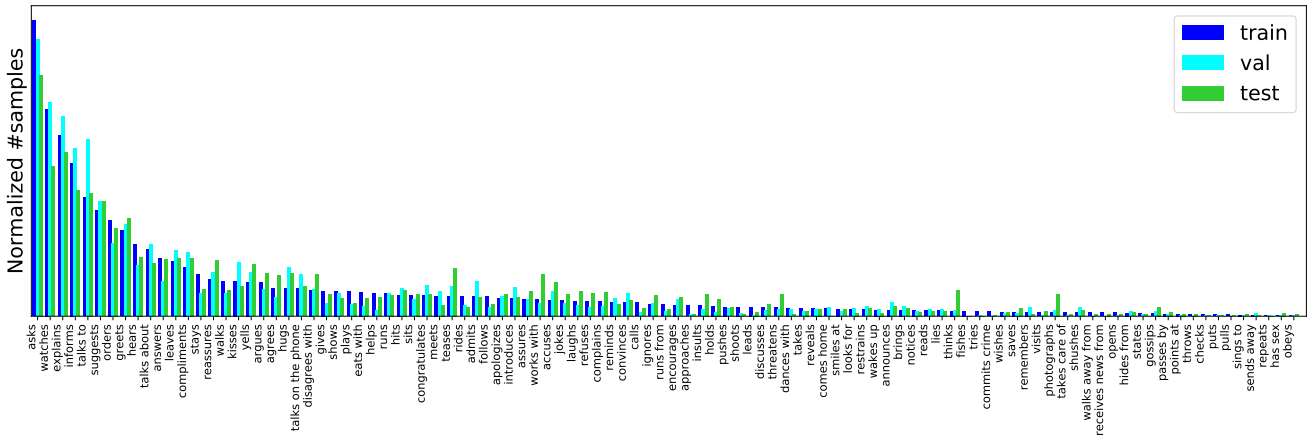

Figure 11: Distribution of interaction labels in train/val/test sets. Sorted by descending order based on train set.

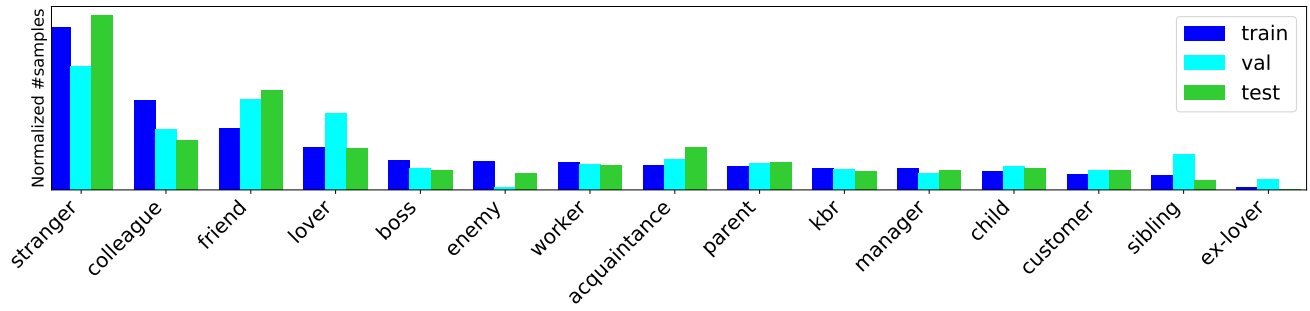

Figure 12: Distribution of relationship labels in train/val/test sets. Sorted by descending order based on train set.

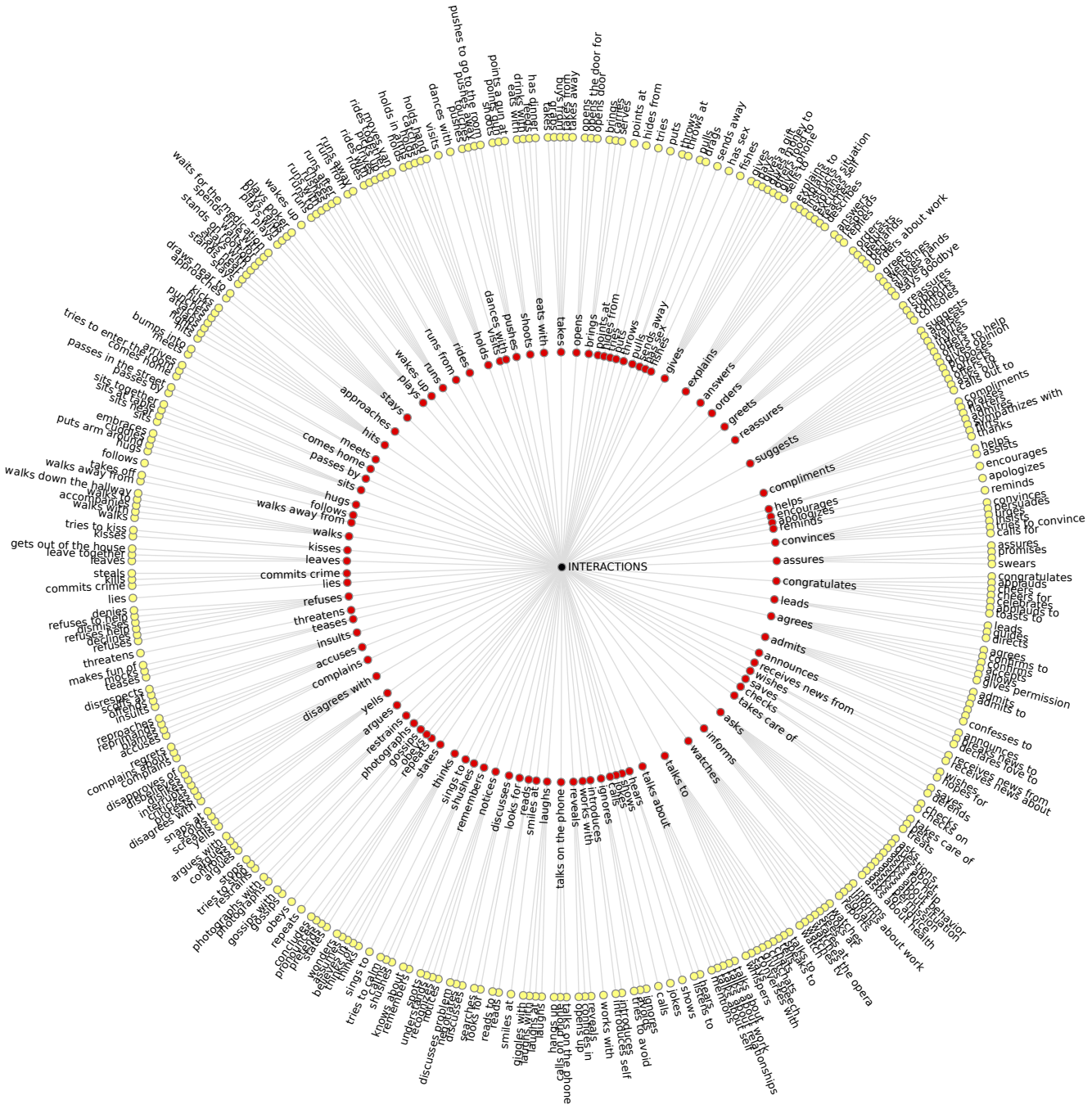

Figure 13: Diagram depicting how we group 324 interaction classes (outer circle) into 101 (inner circle). Best seen on the screen.

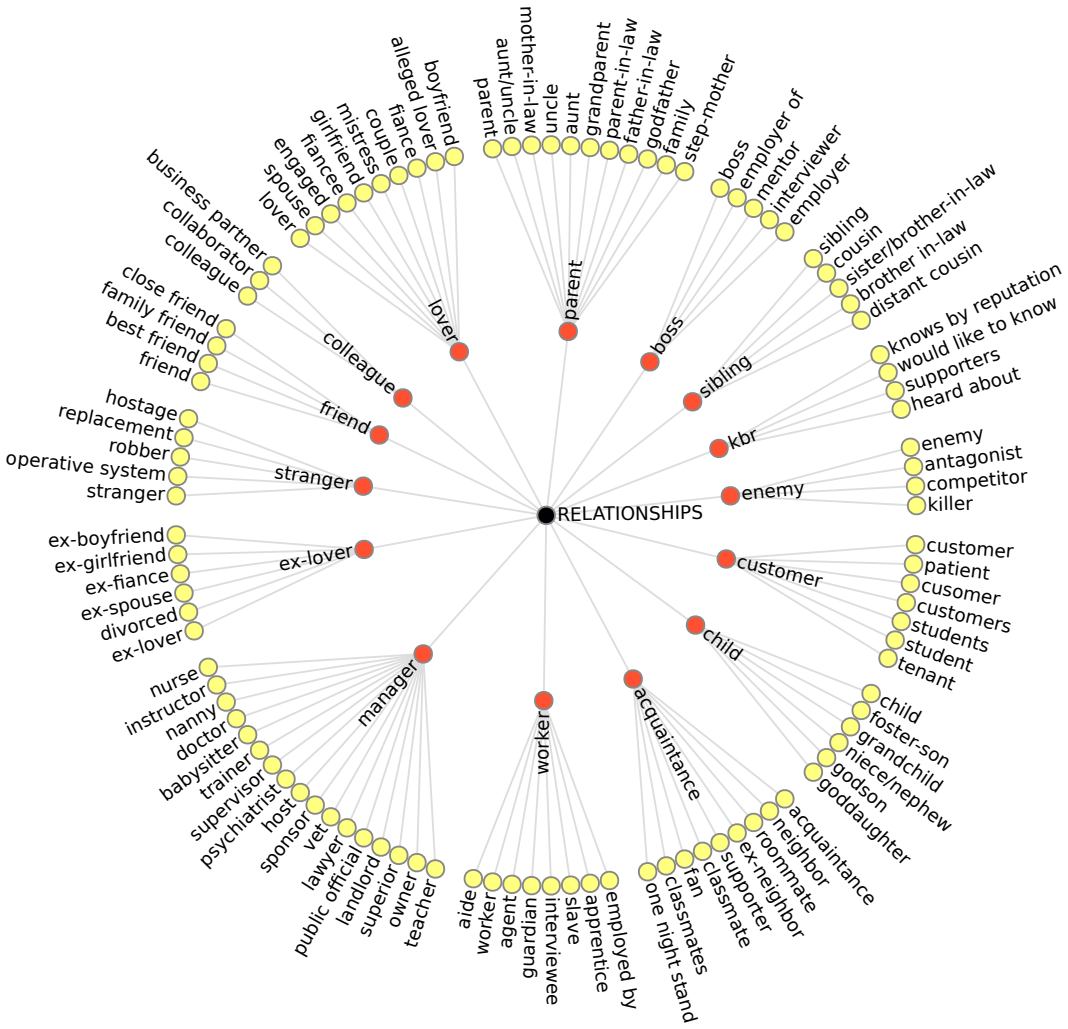

Figure 14: Diagram depicting how we group 107 relationship classes (outer circle) into 15 (inner circle).