Linking People in Videos with "Their" Names Using Coreference Resolution

Vignesh Ramanathan[1], Armand Joulin[2], Percy Liang[2], and Li Fei-Fei[2]

[1] Department of Electrical Engineering, Stanford University, USA
[2] Computer Science Department, Stanford University, USA
{vigneshr,ajoulin,pliang,feifeili}@cs.stanford.edu

Abstract. Natural language descriptions of videos provide a potentially rich and vast source of supervision. However, the highly-varied nature of language presents a major barrier to its effective use. What is needed are models that can reason over uncertainty over both videos and text. In this paper, we tackle the core task of person naming: assigning names of people in the cast to human tracks in TV videos. Screenplay scripts accompanying the video provide some crude supervision about who's in the video. However, even the basic problem of knowing who is *mentioned* in the script is often difficult, since language often refers to people using pronouns (e.g., "he") and nominals (e.g., "man") rather than actual names (e.g., "Susan"). Resolving the identity of these mentions is the task of *coreference resolution*, which is an active area of research in natural language processing. We develop a joint model for person naming and coreference resolution, and in the process, infer a latent alignment between tracks and mentions. We evaluate our model on both vision and NLP tasks on a new dataset of 19 TV episodes. On both tasks, we significantly outperform the independent baselines.

Keywords: Person naming, coreference resolution, text-video alignment.

1 Introduction

It is predicted that video will account for more than 85% of Internet traffic by 2016 [1]. To search and organize this data effectively, we must develop tools that can understand the people, objects, and actions in these videos. One promising source of supervision for building such tools is the large amount of natural language text that typically accompanies videos. For example, videos of TV episodes have associated screenplay scripts, which contain natural language descriptions of the videos (Fig. 1).

In this paper, we tackle the task of *person naming*: identifying the name (from a fixed list) of each person appearing in a TV video. Since the script accompanying a video also mentions these people, we could use the names in the text as labels for person naming. But as seen in Fig. 1, the text does not always use proper names (e.g., "Leonard") to refer to people. Nominal expressions (e.g., 'engineer") and pronouns (e.g., "he") are also employed, accounting for 32% of

D. Fleet et al. (Eds.): ECCV 2014, Part I, LNCS 8689, pp. 95–110, 2014.

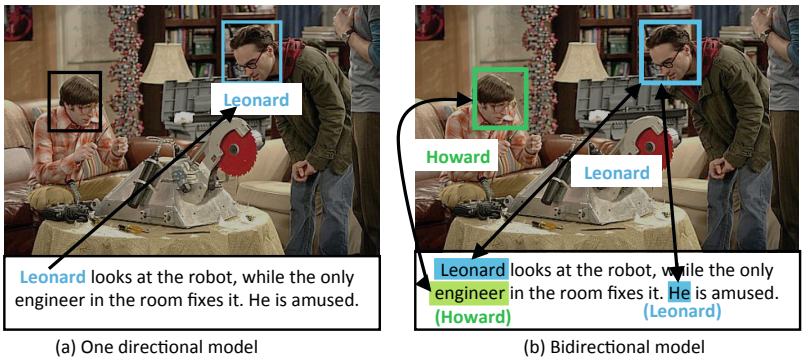


Fig. 1. Name assignment to people in a video can be improved by leveraging richer information from the text. (a) A traditional unidirectional approach only transfers unambiguous mentions of people ("Leonard") from the text to the video. (b) Our proposed bidirectional approach reasons about both proper mentions and ambiguous nominal and pronominal mentions ("engineer", "he").

the human mentions in our dataset. A human reading the text can understand these mentions using context, but this problem of *coreference resolution* remains a difficult challenge and an active area of research in natural language processing [17, 25].

Pioneering works such as [11, 37, 7, 38, 4, 6] sidestep this challenge by only using proper names in scripts, ignoring pronouns and nominals. However, in doing so, they fail to fully exploit the information that language can offer. At the same time, we found that off-the-shelf coreference resolution methods that operate on language alone are not accurate enough. Hence, what is needed is a model that tackles person naming and coreference resolution jointly, allowing information to flow bidirectionality between text and video.

The main contribution of this paper is a new bidirectional model for person naming and coreference resolution. To the best of our knowledge, this is the first attempt that jointly addresses both these tasks. Our model assigns names to tracks and mentions, and constructs an explicit alignment between tracks and mentions. Additionally, we use temporal constraints on the order in which tracks and mentions appear to efficiently infer this alignment.

We created a new dataset of 19 TV episodes along with their complete scripts, collected from 10 different TV shows. On the vision side, our model outperforms unidirectional models [6, 7] in name assignment to human tracks. On the language side, our model outperforms state-of-the-art coreference resolution systems [27, 17] for name assignment to mentions in text.

2 Related Work

Track naming using screenplay scripts. In the context of movies or TV shows, scripts have been used to provide weak labels for person naming [11, 37,

7, 38, 4, 6], and action recognition [26, 28, 9, 6]. All these works use the names from scripts as weak labels in a multiple instance learning (MIL) setting [40]. A similar line of work [32] links person names with faces based on image captions. These methods offer only a unidirectional flow of information from language to vision, and assume very little ambiguity in the text. In contrast, our model propagates information both from text to video (for person naming) and from video to text (for coreference resolution).

Joint vision and language models. Many works have combined NLP and computer vision models; we mention the ones most relevant to our setting. Some focus on creating textual descriptions for images or videos [8, 24, 30, 31, 41, 12, 35]. Others propagate image captions to uncaptioned images using visual similarities [41, 3, 14, 33]. Another line of work focuses on learning classifiers from images with text [5, 15, 21]. Rohrbach et al. [34] introduced semantic relatedness between visual attributes and image classes. Recently, Fidler et al. [13] used image descriptions to improve object detection and segmentation. These methods assume relatively clean text and focus only on propagating information from text to either videos or images. We work in a more realistic setting where text is ambiguous, and we show that vision can help resolve these ambiguities.

Grounding words in image/videos. This is the task of aligning objects or segments in an image/video with corresponding words or phrases in the accompanying text. This problem has been handled in different scenarios, depending on the entity to be grounded: Tellex et al. [39] addressed this challenge in a robotics setting, while others have worked on grounding visual attributes in images [36, 29]. Gupta et al. grounded storylines based on annotated videos [16].

Coreference resolution. Coreference resolution is a core task in the NLP community, and we refer the reader to Kummerfeld et al. [25] for a thorough set of references. Hobbs et al. [19] tried to extend the idea of coreference resolution for entities and events occurring in video transcripts. Hodosh et al. [20] and more recently, Kong et al. [23] have reported improvement in coreference resolution of objects mentioned in a text describing a static scene, when provided with the image of the scene. Unlike these works, we focus on the coreference resolution of humans mentioned in a TV script, where people reappear at multiple time points in the video. In our work, we build a discourse-based coreference resolution model similar to that of Haghighi and Klein [17]. We also take advantage of properties of TV scripts, such as the fixed set of cast names and constraints on the gender of the mentions.

3 Problem Setup

We are given a set of video-script pairs representing one TV episode. Let $\mathcal{P}$ be the set of P names appearing in the cast list, which we assume to be known. We also include a special "NULL" person in $\mathcal{P}$ to represent any person appearing in the episode, but not mentioned in the cast list.

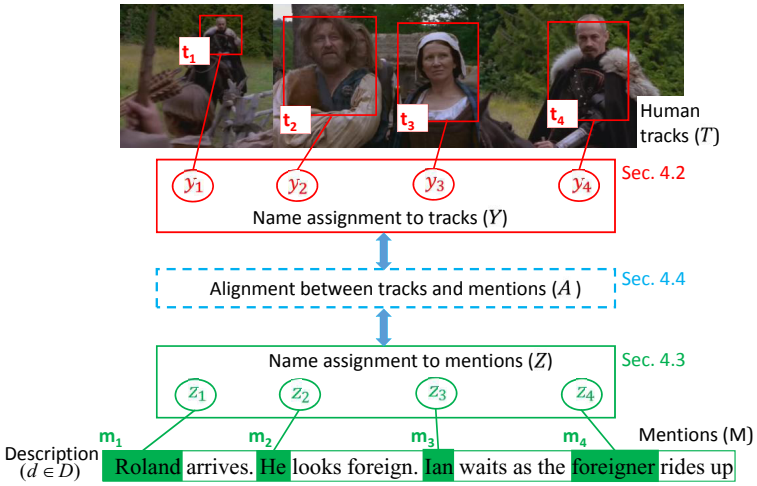


Fig. 2. The problem setup is illustrated for a sample scene. Our task is to assign a person $p \in \mathcal{P}$ to each human track t (red bounding box in the figure) and to each mention m (highlighted in green) from the text.

On the vision side, let $\mathcal{T}$ be a set of T human tracks extracted from the video (see Sec. 6 for details). For each track $t \in \mathcal{T}$, let $y_t \in \mathcal{P}$ denote the person name assigned to track t. We also define a matrix $Y \in \{0,1\}^{T \times P}$, where $Y_{tp} = 1$ iff $y_t = p$.

On the language side, each script is a sequence of *scenes*, and each scene is a sequence of *dialogues* $\mathcal{D}$ and *descriptions* $\mathcal{E}$. From the descriptions, we extract a set $\mathcal{M}$ of M *mentions* corresponding to people (see Sec. 6 for details). A mention is either a proper noun (e.g., "Roland"), pronoun (e.g., "he") or nominal (e.g., "foreigner"). For each mention $m \in \mathcal{M}$, let $z_m \in \mathcal{P}$ denote the person assigned to mention m. Define a matrix $Z \in \{0,1\}^{M \times P}$, where $Z_{mp} = 1$ iff $z_m = p$.

Each dialogue and description is also crudely aligned to a temporal window in the video, using the subtitle-based method from [11]. Our goal is to infer the person assignment matrices Y and Z given the crude alignment, as well as features of the tracks and mentions (see Fig. 2).

4 Our Model

In this section, we describe our model as illustrated in Fig. 2. First, let us describe the variables:

- Name assignment matrix for tracks $Y \in \{0,1\}^{T \times P}$.
- Name assignment matrix for mentions $Z \in \{0,1\}^{M \times P}$.
- Antecedent matrix $R \in \{0,1\}^{M \times M}$ where $R_{mm'}$ indicates whether the mention m (e.g., "he") refers to m' (e.g., "Roland") based on the text (and hence refer to the same person). In this case, m' is called the antecedent of m.

– Alignment matrix $A \in \{0,1\}^{T \times M}$ between tracks and mentions, where A_{tm} indicates whether track t is aligned to mention m.

The first two (Y and Z) are the output variables introduced in the previous section; the other variables help mediate the relationship between Y and Z.

We define a cost function over these variables which decomposes as follows:

$$C(Y, Z, R, A) \stackrel{\text{def}}{=} \gamma_t \cdot C_{\text{track}}(Y) + \gamma_m \cdot C_{\text{mention}}(Z, R) + C_{\text{align}}(A, Y, Z), \quad (1)$$

where γ_t and γ_m are hyperparameters governing the relative importance of each term. The three terms are as follows:

– $C_{\text{track}}(Y)$ is only based on video (face recognition) features (Sec. 4.2).
– $C_{\text{mention}}(Z, R)$ is only based on text features, using coreference features to influence R, and thus the name assignment Z (Sec. 4.3).
– $C_{\text{align}}(A, Y, Z)$ is based on a latent alignment A of the video and which imposes a soft constraint on the relationship between Y and Z (Sec. 4.4).

We minimize a relaxation of the cost function $C(Y, Z, R, A)$; see Sec. 5. Note that we are working in the transductive setting, where there is not a separate test phase.

4.1 Regression-Based Clustering

One of the building blocks of our model is a regression-based clustering method [2]. Given n points $x_1, \ldots, x_n \in \mathbb{R}^d$, the task is to assign, for each point x_i, a binary label vector $y_i \in \{0,1\}^p$ so that nearby points tend to receive the same label. Define the matrix $X = (x_1, \ldots, x_n)^\top \in \mathbb{R}^{n \times d}$ of points and $Y \in \{0,1\}^{n \times p}$ the labels. The regression-based clustering cost function is as follows:

$$\begin{aligned} C_{\text{cluster}}(Y; X, \lambda) &= \arg \min_{\mathbf{W} \in \mathbb{R}^{d \times p}} \sum_{t \in \mathcal{T}} \|Y - X\mathbf{W}\|_F^2 + \lambda \|\mathbf{W}\|_F^2 \quad (2) \\ &= \operatorname{tr}(Y^\top \underbrace{(I - X(X^\top X + \lambda I)^{-1} X^\top)}_{\stackrel{\text{def}}{=} B(X,\lambda)} Y), \end{aligned}$$

where the second line follows by analytically solving for the optimal weights. (see Bach and Harchaoui [2]). Note that if we relax $Y \in \{0,1\}^{n \times p}$ to $Y \in [0,1]^{n \times p}$, then $C_{\text{cluster}}(Y; X, \lambda)$ becomes a convex quadratic function of Y, and can be minimized efficiently. We will use this building block in the next two sections.

4.2 Name Assignment to Tracks

In this section, we describe $C_{\text{track}}(Y)$, which is responsible for the name assignment to tracks based on visual features. Many different models [7, 6, 26, 28] have been proposed to assign names to tracks based on face features. In this work, we adopt the recent model from Bojanowski et al. [6], which was shown to achieve

state-of-the-art performance for this task. Specifically, let $\Phi^{\text{track}} \in \mathbb{R}^{T \times d}$ be a matrix of face features (rows are tracks $t \in \mathcal{T}$ and columns are features). We set $C_{\text{track}}(Y)$ in our cost function (Eq. 1) to be the clustering cost:

$$C_{\text{cluster}}(Y; \Phi^{\text{track}}, \lambda^{\text{track}}) \quad \text{(face features)} \tag{3}$$

We also enforce that each track is be associated with exactly one name: $Y\mathbf{1}_P = \mathbf{1}_T$. This hard constraint (and all subsequent constraints) is included in $C_{\text{track}}(Y)$ by adding a term equal to 0 if the constraint is satisfied and ∞ otherwise.

Additionally, as with standard approaches [7, 6, 26, 28], we include hard constraints based on the crude alignment of the script with the video:

Dialogue alignment constraint. Each dialogue $d \in \mathcal{D}$ is associated with a subset $\mathcal{P}_d$ of speakers, and a subset $\mathcal{T}_d$ of tracks which overlaps with the dialogue. This overlap is obtained from the crude alignment between tracks and dialogues [11]. Similar to [6], we add a *dialogue alignment* constraint enforcing that each speaker in $\mathcal{P}_d$ should align to at least one track in $\mathcal{T}_d$.

$$\forall\ d \in \mathcal{D},\ \forall\ p \in \mathcal{P}_d : \sum_{t \in \mathcal{T}_d} Y_{tp} \geq 1 \quad \text{(dialogue alignment)} \tag{4}$$

Scene alignment constraint. Each scene $s \in \mathcal{S}$ is associated with a subset of names $\mathcal{P}_s$ mentioned in the scene, and a subset of tracks $\mathcal{T}_s$ which overlaps with the scene (also from crude alignment [11]). We observe in practice that *only* names in $\mathcal{P}_s$ appear in the scene, so we add a *scene alignment* constraint enforcing that a name *not mentioned* in scene s should not be aligned to a track in $\mathcal{T}_s$:

$$\forall s \in \mathcal{S},\ p \notin \mathcal{P}_s : \sum_{t \in \mathcal{T}_s} Y_{tp} = 0 \quad \text{(scene alignment)} \tag{5}$$

Note that this constraint was absent from [6].

4.3 Name Assignment to Mentions and Coreference Resolution

In this section, we describe $C_{\text{mention}}(Z, R)$, which performs name assignment to mentions. The nature of name assignment to mentions is notably different from that of tracks. Proper mentions such as "Roland" are trivial to map to the fixed set of cast names based on string match, but nominal (e.g., "foreigner") and pronominal (e.g., "he") mentions are virtually impossible to assign based on the mention alone. Rather, these mentions reference previous *antecedent* mentions (e.g., "Roland" in Fig. 3). The task of determining antecedent links is called coreference resolution in NLP [25].[1]

To perform coreference resolution, we adapt the discourse model of [17], retaining their features, but using our clustering framework (Sec. 4.1).

[1] Our setting differs slightly from classic coreference resolution in that we must resolve each mention to a fixed set of names, which is the problem of *entity linking* [18].

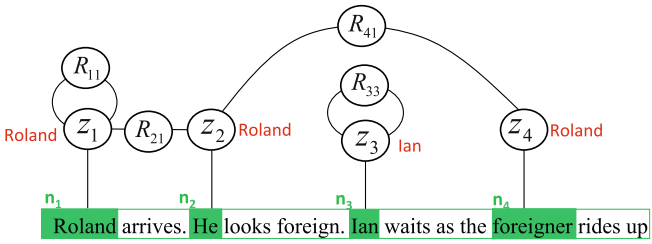


Fig. 3. An example illustrating the mention naming model from Sec. 4.3. The mentions in the sentence are highlighted in green. The antecedent variable R and name assignment matrix Z are shown for the correct coreference links. The final names assigned to the mentions are shown in red.

Coreference resolution. Each pair of mentions (m, m') is represented by a d-dimensional feature vector, and let $\Phi^{\text{mention}} \in \mathbb{R}^{M^2 \times d}$ be the corresponding feature matrix.[2] We apply the clustering framework (Sec. 4.1) to predict the antecedent matrix, or more precisely, its vectorized form $\text{vec}(R) \in \mathbb{R}^{M^2}$. We first include in $C_{\text{mention}}(R, Z)$ (Eq. 1) the clustering cost:

$$C_{\text{cluster}}(\text{vec}(R); \Phi^{\text{mention}}, \lambda^{\text{mention}}) \quad \text{(coreference features)} \tag{6}$$

We also impose hard constraints, adding them to $C_{\text{mention}}(R, Z)$. First, each mention has at most one antecedent:

$$\forall\, m \leq M : \sum_{m' \leq m} R_{mm'} = 1 \quad \text{(one antecedent)} \tag{7}$$

In addition, we include linguistic constraints to ensure gender consistency and to avoid self-association of pronouns (see supplementary material for the details).

Connection constraint. When m has an antecedent m' ($R_{mm'} = 1$), they should be assigned the same name ($Z_m = Z_{m'}$). Note that the converse is not necessarily true: two mentions not related via the antecedent relation ($R_{mm'} = 0$) can still have the same name. For example, in Fig. 3, the mentions "Roland" and "foreigner" are not linked, but still refer to the same person. This relation between Z and R can be enforced through the following constraint:

$$\forall\, m' \leq m, \forall\, p \in \mathcal{P} : \; |Z_{mp} - Z_{m'p}| \leq 1 - R_{mm'} \quad (R \text{ constrains } Z) \tag{8}$$

Finally, each mention is assigned exactly one name: $Z\mathbf{1}_P = \mathbf{1}_M$.

4.4 Alignment between Tracks and Mentions

So far, we have defined the cost functions for the name assignment matrices for tracks Y and mentions Z, which use video and text information separately.

[2] The features are described in [17]. They capture agreement between different attributes of a pair of mentions, such as the gender, cardinality, animacy, and position in the parse tree.

Now, we introduce $C_{\text{align}}(A, Y, Z)$, the alignment part of the cost function, which connects Y and Z, allowing information to flow between text and video.

There are three intuitions involving the alignment matrix A: First, a track and a mention that are aligned should be assigned the same person. Second, the tracks $\mathcal{T}$ and mentions $\mathcal{M}$ are ordered sequences, and an alignment between them should be monotonic. Third, tracks and mentions that occur together based on the crude alignment ([11]) are more likely to be aligned. We use these intuitions to formulate the alignment cost $C_{\text{align}}(A, Y, A)$, as explained below:

Monotonicity constraint. The tracks $\mathcal{T}$ are ordered by occurrence time in the video, and the mentions $\mathcal{M}$ are ordered by position in the script. We enforce that no alignment edges cross (this assumption is generally but not always true): if $t_2 > t_1$ and $A_{t_1 m} = 1$, then $A_{t_2 m'} = 0$ for all $m' < m$.

Mention mapping constraint. Let $\mathcal{M}_e$ be the set of mentions in a description $e \in \mathcal{E}$ and $\mathcal{T}_e$ be the set of tracks in the crudely-aligned time window. We enforce each mention from $\mathcal{M}_e$ to be mapped to exactly one track from $\mathcal{T}_e$: for each $e \in \mathcal{E}$ and $m \in \mathcal{M}_e$, $\sum_{t \in \mathcal{T}_e} A_{tm} = 1$. Conversely, we allow a track to align to multiple mentions. For example, in "John sits on the chair, while he is drinking his coffee", a single track might align to both "John" and "he".

$$\forall\ e \in \mathcal{E},\ m \in \mathcal{M}_e,\ \sum_{t \in \mathcal{T}_e} A_{tm} = 1 \quad \text{(mention mapping)}. \tag{9}$$

Connection penalty. If a track t is assigned to person p ($Y_{tp} = 1$), and track t is aligned to mention m ($A_{tm} = 1$), then mention m should be assigned to person p as well ($Z_{mp} = 1$). To enforce this constraint in a soft way, we add the following penalty:

$$\|A^\top Y - Z\|_F^2 = -2\text{tr}(A^\top Y Z) + \text{constant}, \tag{10}$$

where the equality leverages the fact that Y and Z are discrete with rows that sum to 1 (see supplementary material for details). Note that $C_{\text{align}}(A, Y, Z)$ is thus a linear function of A with monotonicity constraints. This special form will be important for optimization in Sec. 5.

5 Optimization

Now we turn to optimizing our cost function (Eq. 1). First, the variables Z, Y, R and A are matrices with values in $\{0, 1\}$. We relax the domains of all variables except A from $\{0, 1\}$ to $[0, 1]$. Additionally, to account for noise in the tracks and mentions, we add a slack to all inequalities involving Y and Z.

We solve the relaxed optimization problem using block coordinate descent, where we cycle between minimizing Y, (Z, R), and A. Each block is convex given the other blocks. For the smaller matrices Z, Y and R (which have on

the order of 10^4 elements), we use interior-point methods [6], whose complexity is cubic in the number of variables. The alignment matrix A has on the order of 10^6 elements, but fortunately, due to the special form of $C_{\text{align}}(A, Y, Z)$, we can use an efficient dynamic program similar to dynamic time warping [10] to optimize A (see supplementary material for details).

Initialization. Since our cost function is not jointly convex in all the variables, initialization is important. We initialize our method with the solution to simplified optimization problem that excludes any terms involving more than one block of variables.

Rounding. The variables Y and Z are finally rounded to integer matrices, with elements in $\{0, 1\}$. The rounding is carried out similar to [6], by projecting the matrices on the corresponding set of integer matrices. This amounts to taking the maximum value along the rows of Y and Z.

6 Experiments

We evaluated our model on the two tasks: (i) name assignment to tracks in videos and (ii) name assignment to mentions in the corresponding scripts.

Dataset. We created a new dataset of TV episode videos along with their scripts.[3] Previous datasets [6, 7, 38] come with heavily preprocessed scripts, where no ambiguities in the text are retained. In contrast, we use the original scripts. We randomly chose 19 episodes from 10 different TV shows. The complete list of the episodes is shown in the supplementary material. Sample video clips from the dataset with descriptions are shown in Fig. 4. The dataset is split into a development set of 14 episodes and a test set of 5 episodes. Note that there is no training set, as we are working in the transductive setting. The number of names in the cast lists varies between $9 - 21$.

To evaluate the name assignment task in videos, we manually annotated the names of human tracks from 3 episodes of the development set, and all 5 episodes of the test set. There are a total of 3329 tracks with ground truth annotations in the development set, and 4757 tracks in the test set. To evaluate the name assignment to mentions, we annotated the person names of the pronouns and nominal mentions in all episodes. To ensure that a mention always refers to a person physically in the scene, we retain only the mentions which are the subject of a verb. This resulted in a total of 811 mentions in the development set and 300 mentions in the test set.

Implementation details. The tracks were obtained by running an off-the-shelf face detector followed by tracking [22]. We retain all tracks extracted by this scheme, unlike previous works which only use a subset of clean tracks with

[3] The scripts were collected from `https://sites.google.com/site/tvwriting/`.

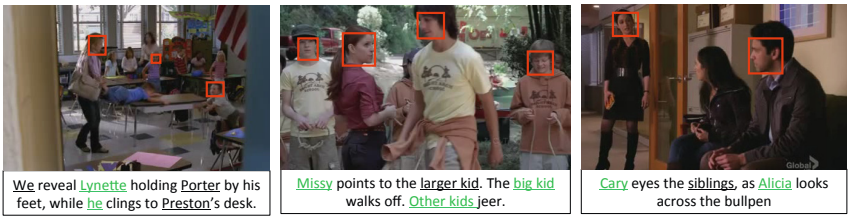


Fig. 4. Sample video clips from the dataset are shown along with their corresponding script segments. The mentions extracted from the script are underlined. The ones corresponding to nominal subjects are shown in green. These are the mentions used in our full model for person name assignment. The face tracks from the video are shown by red bounding boxes.

visible facial features. We further extracted a set of features between pairs of mentions using the Stanford CoreNLP toolbox [27] (see supplementary material). We tuned the hyperparameters on the development set, yielding $\lambda^{\text{mention}} = 0.0001$, $\lambda^{\text{track}} = 0.01$, $\gamma_t = 0.2$ and $\gamma_m = 20$.

Table 1. The Average Precision (AP) scores for person name assignment in videos is shown for episodes with face annotations in the development and test set. We also show the mean AP (MAP) value for the development and test sets. The description of the different methods are provided in the text.

Set	Development				Test					
Episode ID	E1	E2	E3	*MAP*	E15	E16	E17	E18	E19	*MAP*
RANDOM	0.266	0.254	0.251	0.257	0.177	0.217	0.294	0.214	0.247	0.229
COUR [7]	0.380	0.333	0.393	0.369	0.330	0.327	0.342	0.306	0.337	0.328
BOJ [6]	0.353	0.434	0.426	0.404	0.285	0.429	0.378	0.383	**0.454**	0.385
OURUNIDIR	0.512	0.560	0.521	0.531	0.340	0.474	0.503	0.399	0.384	0.420
OURUNICOR	0.497	0.572	0.501	0.523	**0.388**	0.470	0.512	0.424	0.401	0.431
OURUNIF	0.497	0.552	0.561	0.537	0.345	0.488	0.516	0.410	0.388	0.429
OURBIDIR	**0.567**	**0.665**	**0.573**	**0.602**	0.358	**0.518**	**0.587**	**0.454**	0.376	**0.459**

6.1 Name Assignment to Tracks in Video

We use the Average Precision (AP) metric previously used in [6, 7] to evaluate the performance of person naming in videos. We compare our model (denoted by OURBIDIR) to state-of-the-art methods and various baselines:

1. RANDOM: Randomly picks a name from the set of possible names consistent with the crude alignment.
2. BOJ [6]: Similar to the model described in Sec. 4.2 but without the scene constraints. We use the publicly available code from the authors.

3. COUR [7]: Weakly-supervised method for name assignment using a discriminative classifier. We use the publicly available code from the authors.
4. OURUNIDIR: Unidirectional model which does not use any coreference resolution, but unlike BOJ, it includes the "scene alignment" constraint.
5. OURUNICOR: We first obtain the person names corresponding to the mentions in the script by running our coreference model from Sec. 4.3. These are then used to specify additional constraints similar to the "dialogue alignment" constraint.
6. OURUNIF: All the tracks appearing in the temporal window corresponding to the mention are given equal values in the matrix A.
7. OURBIDIR: Our full model which jointly optimizes name assignment to mentions and tracks.

Tab. 1 shows the results. First, note that even our unidirectional model (OURUNIDIR) performs better than the state-of-the-art methods from [6, 7]. As noted in [6], the ridge regression model from Bach et al [2] might be more robust to noise. This could explain the performance gain over [7]. The improvement of our unidirectional model over [6] is due to our addition of scene based constraints, which reduces the ambiguity in the names that can be assigned to a track.

The improved performance of our bidirectional model compared to OURUNIDIR and OURUNICOR, shows the importance of the alignment variable in our formulation. On the other hand, when A is fixed through uniform assignment (OURUNIF), the model performs worse than our bidirectional model. This shows the benefit of inferring the alignment variable in our method.

In Fig. 5, we show examples where our model makes correct name assignments. Here, our model performs well even when tracks are aligned with pronouns and nominals.

Finally, we conducted an oracle experiment where we fix the alignment between tracks and mentions (A) and mention name assignment (Z) to manually-annotated ground truth values. The resulting OURBIDIR obtained a much improved MAP of 0.565 on the test set. We conclude that person naming could be improved by inferring a better alignment variable.

6.2 Name Assignment to Mentions

Now we focus on the language side. Here, our evaluation metric is accuracy, the fraction of mentions that are assigned the correct person name. We compare the performance of our full bidirectional model (OURBIDIR) with standard coreference resolution systems and several baselines:

1. CORENLP: This is the coreference resolution model used in the Standord CoreNLP toolbox [27].
2. HAGHIGHI ([17] modified): We modify the method from [17] to account for the fixed set of cast names in our setting (see supplementary material for more details).

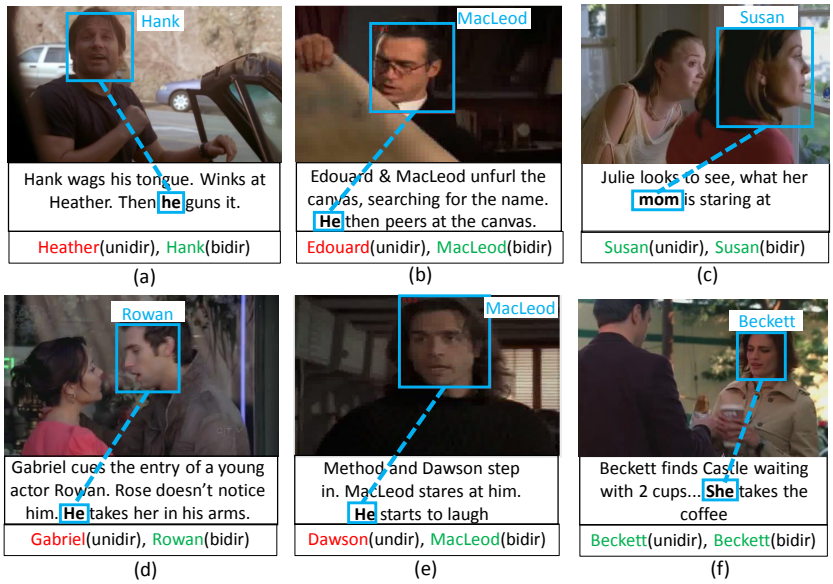


Fig. 5. Examples where the person name assignment by our bidirectional model is correct, both in the video and the text. The alignment is denoted by the blue dotted line. The name assignment to the tracks are shown near the bounding box. The name assignment to mentions by our unidirectional and bidirectional models are shown in the box below the videos. The correct assignment is shown in green, and the wrong one in red.

3. OurUnidir: This is the unidirectional model from Sec. 4.3.
4. OurUnif: Same as OurUnif for name assignment to tracks.
5. OurBidir: Same as OurBidir for name assignment to tracks.

The Stanford CoreNLP coreference system uses a fixed set of rules to iteratively group mentions with similar properties. These rules were designed for use with well structured news articles which have a higher proportion of nominals compared to pronouns. Also, our model performs explicit entity linking by associating every mention to a name from the cast list, unlike standard coreference resolution methods. While comparing to CoreNLP, we performed entity linking by assigning each mention in a coreference chain to the head mention of the chain, which is usually a proper noun corresponding to one of the cast names. These factors contribute to the gain of OurUnidir, which uses constraints specific to the TV episode setting. The modified version of Haghighi and Klein's model [17] is a probabilistic variant of OurUnidir. Note that our formulation is convex whereas theirs is not.

We also a gain from our bidirectional model over the unidirectional model, due to additional visual cues. This is especially true when there text is truly ambiguous. In Fig. 5(d), "Rowan" is not the subject of the sentence preceding the pronoun "He". This causes a simple unidirectional model to associate the

Table 2. The percentage accuracy of mentions associated to the correct person name across all episodes in the development and test set is shown. The description of the different methods in the table are provided in the text.

Set	Dev.	Test
CoreNLP [27]	54.99 %	41.00 %
Haghighi [17] modified	53.02 %	38.67 %
OurUnidir	58.20 %	49.00 %
OurUnif	59.56 %	48.33 %
OurBidir	**60.42 %**	**56.00 %**

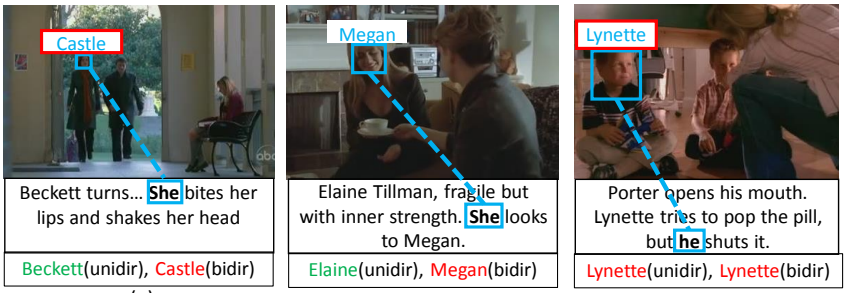
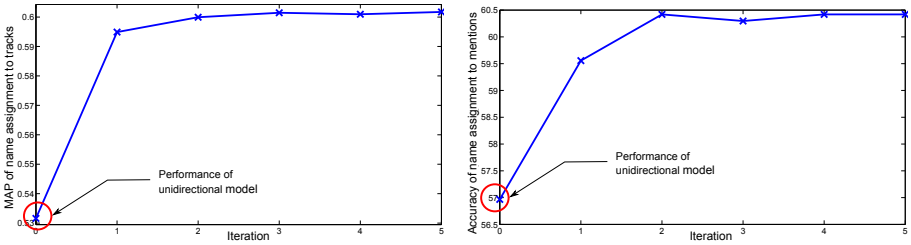


Fig. 6. Examples of videos are shown, where our full model fails to predict the correct names The alignment is shown by the blue dotted line. The name assignment to tracks are shown over the bounding box. The wrong name assignments are shown in red. The name assignment to mentions by our bidirectional and unidirectional models are shown in the box below the videos. The correct name assignment is shown in green, and the wrong one in red.

pronoun with a wrong antecedent mention. Our bidirectional model avoids these errors by using the name of the tracks mapped to the mentions.

Finally, we explore the full potential of improving mention name assignment from visual cues by fixing the matrices A and Y to their ground truth values. This yields an oracle accuracy of 68.98% on the test data, compared to 52.15% for OurBidir. Interestingly, the oracle improvement here on the language side is significantly higher than on the vision side.

Error analysis. We show sample video clips in Fig. 6, where our bidirectional model (OurBidir) fails to predict the correct name for mentions. As seen in Fig. 6(a), one typical reason for failure is incorrect name assignments to low-resolution faces; the error then propagates to the mentions. In the second example, the face detector fails to capture the face of the person mentioned in the script. Hence, our model maps the pronoun to the only face available in the description, which is incorrect.



(a) Name assignment to tracks (b) Name assignment to mentions

Fig. 7. (a) Mean average precision (MAP) of person naming of tracks at different iterations. (b) Accuracy of person naming of mentions at different iterations.

Empirical justification of our joint optimization. We also show the performance of our full model at each iteration. Fig. 7 plots the MAP for person naming and accuracy for coreference resolution on the development set. We observe that performance on both tasks jointly improves over time, showing the importance of a bidirection flow of information between text and video.

7 Conclusion

In this work, we tackled the problem of name assignment to people in videos based on their scripts. Compared to previous work, we leverage richer information from the script by including ambiguous mentions of people such as pronouns and nominals. We presented a bidirectional model to jointly assign names to the tracks in the video and the mentions in the text; a latent alignment linked the two tasks. We evaluated our method on a new dataset of 19 TV episodes. Our full model provides a significant gain for both vision and language tasks compared to models that handle the tasks independently. We plan to extend our bidirectional model to not only share information about the identity of the tracks and mentions, but also to link the actions in video with relations in text.

Acknowledgements. We thank A. Fathi, O. Russakovsky and S. Yeung for helpful comments and feedback. This research is partially supported by Intel, the NFS grant IIS-1115493 and DARPA-Mind's Eye grant.

References

1. Cisco visual networking index: Global mobile data traffic forecast update. Tech. rep., Cisco (February 2014)
2. Bach, F., Harchaoui, Z.: Diffrac: A discriminative and flexible framework for clustering. In: NIPS (2007)

3. Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D.M., Jordan, M.I.: Matching words and pictures. Journal of Machine Learning Research 3, 1107–1135 (2003)
4. Bäuml, M., Tapaswi, M., Stiefelhagen, R.: Semi-supervised Learning with Constraints for Person Identification in Multimedia Data. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (June 2013)
5. Berg, T.L., Berg, A.C., Edwards, J., Maire, M., White, R., Teh, Y.W., Learned-Miller, E.G., Forsyth, D.A.: Names and faces in the news. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 848–854 (2004)
6. Bojanowski, P., Bach, F., Laptev, I., Ponce, J., Schmid, C., Sivic, J.: Finding actors and actions in movies. In: ICCV (2013)
7. Cour, T., Sapp, B., Jordan, C., Taskar, B.: Learning from ambiguously labeled images. In: CVPR (2009)
8. Das, P., Xu, C., Doell, R.F., Corso, J.J.: A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In: CVPR (2013)
9. Duchenne, O., Laptev, I., Sivic, J., Bach, F., Ponce, J.: Automatic annotation of human actions in video. In: ICCV (2009)
10. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. John Wiley & Sons (2012)
11. Everingham, M., Sivic, J., Zisserman, A.: Hello! my name is... buffy automatic naming of characters in tv video. In: BMVC (2006)
12. Farhadi, A., Hejrati, M., Sadeghi, M., Young, P., Rashtchian, C., Hockenmaier, J., Forsyth, D.: Every picture tells a story: Generating sentences from images. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 15–29. Springer, Heidelberg (2010)
13. Fidler, S., Sharma, A., Urtasun, R.: A sentence is worth a thousand pixels. In: CVPR. IEEE (2013)
14. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In: CVPR (2009)
15. Gupta, A., Davis, L.S.: Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 16–29. Springer, Heidelberg (2008)
16. Gupta, A., Srinivasan, P., Shi, J., Davis, L.S.: Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 2012–2019. IEEE (2009)
17. Haghighi, A., Klein, D.: Coreference resolution in a modular, entity-centered model. In: HLT-NAACL (2010)
18. Han, X., Sun, L., Zhao, J.: Collective entity linking in web text: a graph-based method. In: ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 765–774 (2011)
19. Hobbs, J.R., Mulkar-Mehta, R.: Using abduction for video-text coreference. In: Proceedings of BOEMIE 2008 Workshop on Ontology Evolution and Multimedia Information Extraction (2008)
20. Hodosh, M., Young, P., Rashtchian, C., Hockenmaier, J.: Cross-caption coreference resolution for automatic image understanding. In: Conference on Computational Natural Language Learning (2010)
21. Jie, L., Caputo, B., Ferrari, V.: Who's doing what: Joint modeling of names and verbs for simultaneous face and pose annotation. In: NIPS (2009)

22. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 34(7), 1409–1422 (2012)
23. Kong, C., Lin, D., Bansal, M., Urtasun, R., Fidler, S.: What are you talking about? text-to-image coreference. In: CVPR (2014)
24. Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A.C., Berg, T.L.: Baby talk: Understanding and generating simple image descriptions. In: CVPR (2011)
25. Kummerfeld, J.K., Klein, D.: Error-driven analysis of challenges in coreference resolution. In: Proceedings of EMNLP (2013)
26. Laptev, I., Marszałek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR (2008)
27. Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., Jurafsky, D.: Stanford's mulit-pass sieve coreference resolution system at the conll-2011 shared task. In: CoNLL 2011 Shared Task (2011)
28. Marszałek, M., Laptev, I., Schmid, C.: Actions in context. In: CVPR (2009)
29. Matuszek, C., FitzGerald, N., Zettlemoyer, L., Bo, L., Fox, D.: A joint model of language and perception for grounded attribute learning. In: ICML (2012)
30. Motwani, T.S., Mooney, R.J.: Improving video activity recognition using object recognition and text mining. In: ECAI (2012)
31. Ordonez, V., Kulkarni, G., Berg, T.L.: Im2text: Describing images using 1 million captioned photographs. In: NIPS (2011)
32. Pham, P., Moens, M.F., Tuytelaars, T.: Linking names and faces: Seeing the problem in different ways. In: Proceedings of the 10th European Conference on Computer Vision: Workshop Faces in 'Real-life' Images: Detection, Alignment, and Recognition, pp. 68–81 (2008)
33. Ramanathan, V., Liang, P., Fei-Fei, L.: Video event understanding using natural language descriptions. In: ICCV (2013)
34. Rohrbach, M., Stark, M., Szarvas, G., Schiele, B.: What helps where – and why? semantic relatedness for knowledge transfer. In: CVPR (2010)
35. Rohrbach, M., Wei, Q., Titov, I., Thater, S., Pinkal, M., Schiele, B.: Translating video content to natural language descriptions. In: ICCV (2013)
36. Silberer, C., Ferrari, V., Lapata, M.: Models of semantic representation with visual attributes. In: ACL (2013)
37. Sivic, J., Everingham, M., Zisserman, A.: "Who are you?" - learning person specific classifiers from video. In: CVPR (2009)
38. Tapaswi, M., Bäuml, M., Stiefelhagen, R.: "Knock! Knock! Who is it?" Probabilistic Person Identification in TV Series. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (June 2012)
39. Tellex, S., Kollar, T., Dickerson, S., Walter, M.R., Banerjee, A.G., Teller, S., Roy, N.: Understanding natural language commands for robotic navigation and mobile manipulation. AAAI (2011)
40. Vijayanarasimhan, S., Grauman, K.: Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
41. Wang, Y., Mori, G.: A discriminative latent model of image region and object tag correspondence. In: NIPS (2010)