Identification of Alias Links among Participants in Narratives

Sangameshwar Patil* **Sachin Pawar*** **Swapnil Hingmire*** **Girish K. Palshikar**
{sangameshwar.patil,sachin7.p}@tcs.com
{swapnil.hingmire,gk.palshikar}@tcs.com
TCS Research, Tata Consultancy Services, India

Vasudeva Varma
vv@iiit.ac.in
IIIT Hyderabad, India

Pushpak Bhattacharyya
pb@cse.iitb.ac.in
IIT Patna, India

Abstract

Identification of distinct and independent participants (entities of interest) in a narrative is an important task for many NLP applications. This task becomes challenging because these participants are often referred to using multiple aliases. In this paper, we propose an approach based on linguistic knowledge for identification of aliases mentioned using proper nouns, pronouns or noun phrases with common noun headword. We use Markov Logic Network (MLN) to encode the linguistic knowledge for identification of aliases. We evaluate on four diverse history narratives of varying complexity as well as newswire subset of ACE 2005 dataset. Our approach performs better than the state-of-the-art.

1 Introduction

Identifying aliases of participants in a narrative is crucial for many NLP applications like timeline creation, question-answering, summarization, and information extraction. For instance, to answer a question (in the context of Table 1) *When did Napoleon defeat the royalist rebels?*, we need to identify `Napoleon` and `the young lieutenant` as aliases of `Napoleon Bonaparte`. Similarly, timeline for `Napoleon Bonaparte` will be inconsistent with the text, if `the young lieutenant` is not identified as an alias `Napoleon Bonaparte`. This will further affect any analysis of the timeline (Bedi et al., 2017).

In the context of narrative analysis, we define –
• A *participant* as an entity of type PERSON (PER), LOCATION (LOC), or ORGANIZATION (ORG). A participant has a *canonical mention*, which is a standardized reference to that participant (e.g., `Napoleon Bonaparte`). Further, it may have several *aliases*, which are different mentions referring to the same participant.
• A *basic participant mention* can be a sequence of proper nouns (e.g., `Napoleon` or `N. Bonaparte`), a pronoun (e.g., `he`) or a *generic NP*[1] (e.g., `a short man` or `the young lieutenant`).
• *Independent basic mentions* of a participant play primary role in the narrative. *Dependent basic mentions* play supporting role by qualifying or elaborating independent basic mentions. For each independent mention, we merge all its dependent mentions to create its *composite mention*.

Note that our notion of dependency is syntactic. A basic mention can be either dependent or independent. A basic mention is said to be *dependent* if its governor in the dependency parse tree is itself a participant mention; otherwise it is called as *independent* mention. An independent mention can be a basic (if it does not have any dependent mentions) or a composite mention. An in-

[Napoleon Bonaparte]$_{P1}$ was quite [a short man]$_{A1}$ just five feet three inches tall. When [he]$_{A1}$ was nine years old, [his parents]$_{P2}$ sent [him]$_{A1}$ to [a military school in France]$_{P3}$. In 1785, [he]$_{A1}$ became [a lieutenant]$_{A1}$. When the Revolution broke out, [Napoleon]$_{A1}$ joined [the army of the new government]$_{P4}$. When [royalist rebels]$_{P5}$ marched on [the National Convention]$_{P6}$, [a government official]$_{P7}$ told [the young lieutenant]$_{A1}$ to defend [the delegates]$_{P8}$.

Table 1: Example narrative excerpt with only independent participant mentions marked. For the i-th participant, canonical mention is marked with Pi and all its aliases are marked with Ai.

*These authors contributed equally.

[1]NP with a common noun headword

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 63–68
Melbourne, Australia, July 15 - 20, 2018.

dependent composite mention is created by recursively merging all its dependent mentions. For instance, for the phrases `his parents` and `parents of Napoleon`, following are the basic participant mentions - `his`, `Napoleon`, and `parents`. In the dependency parse trees, `parents` is the governor in both cases. Hence, `his` and `Napoleon` would be basic dependent mentions. Final independent composite mentions `his parents` or `parents of Napoleon` are created by merging the dependent mentions with the independent mention `parents`.

In this paper, we focus on identification of independent mentions (basic as well as composite) for any participant in a narrative. The problem of identifying aliases of participants is challenging because even though the standard NLP toolkits work well to resolve the coreferences among pronouns and named entities, we observed that they perform poorly for generic NPs. For instance, Stanford CoreNLP does not identify `the young lieutenant` and `Napoleon Bonaparte` as the same participant (Table 1); a task we aim to do. This task can be considered as a sub-problem of the standard coreference resolution (Ng, 2017). We build upon output from any standard coreference resolution algorithm, and improve it significantly to detect the missing aliases.

Our goal is to identify the canonical mentions of all independent participants and their aliases. In this paper, we propose a linguistically grounded algorithm for alias detection. Our algorithm utilizes WordNet hypernym structure for identifying participant mentions. It encodes linguistic knowledge in the form of first order logic rules and performs inference in Markov Logic Networks (MLN) (Richardson and Domingos, 2006) for establishing alias links among these mentions.

2 Related Work

Traditionally, alias detection restricts the focus on aliases of named entities which occur as proper nouns (Sapena et al., 2007; Hsiung et al., 2005) using lexical, semantic, and social network analysis. This ignores the aliases which occur as generic NPs. Even in the coreference resolution, recently (Peng et al., 2015a,b) the focus has come back to generic NP aliases by detecting mention heads. Peng et al. (2015b) propose a notion of *Predicate Schemas* to capture interaction between entities at predicate level and instantiate them using knowledge sources like Wikipedia. These instances of Predicate Schemas are then compiled into constraints in an Integer Linear Programming (ILP) based formulation to resolve coreferences. In addition to pronouns, our approach focuses on identification of common noun based aliases of a participant using MLN.

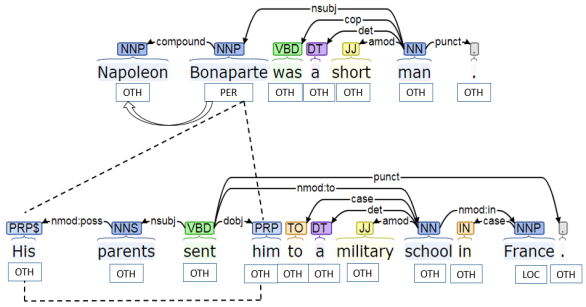

Figure 1: Input ULDG initialized with NER + Coreference. (Note: alias edges(E_a) are shown using dotted lines; participant edges (E_p) are shown using thick arrows; dependency edges (E_d) are shown using thin labelled arrows.)

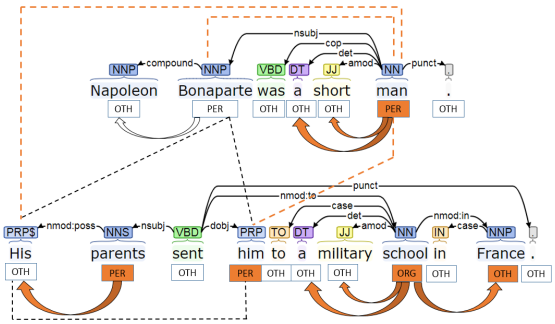

Figure 2: Output ULDG after applying Algorithm 1 on input ULDG in Figure 1. New E_a edges: ⟨`man, Bonaparte`⟩, ⟨`man, him`⟩, & ⟨`man, His`⟩ are added. Newly added E_p edges are highlighted with thick, filled arrows. Participant types of `man` & `school` are changed to PER & ORG respectively; type of `France` is changed to OTH.

MLN has been used to solve the problem of coreference resolution (Poon and Domingos, 2008; Song et al., 2012). Our work differs from them as we build upon output of off-the-shelf coreference resolution system, rather than identifying aliases/coreferences from scratch. This helps in exploiting the strengths (such as linking pronoun mentions to their antecedents) of the existing systems and overcome the weaknesses (such as resolving generic NP mentions) by incorporating additional linguistic knowledge.

A more general and challenging problem in-

volves resolution of bridging descriptions which study relationships between a definite description and its antecedent. As noted in (Vieira and Teufel, 1997; Poesio et al., 1997), bridging descriptions consider many different types of relationships between a definite description (definite generic NP) and its antecedent; e.g., synonymy, hyponymy, meronymy, events, compound nouns, etc. However, in this paper we focus on *identity* type of relationships only. Further, Vieira and Teufel (1997) use WordNet to identify these relationship types between definite descriptions. As described in Phase-I of algorithm 1 (Section 3), we use WordNet for a completely different purpose of identifying participant type.[2] Gardent and Kow (2003) presented a corpus study of bridging definite descriptions and their typologies. They have identified several types of bridging relations like set-subset, event-argument etc.

3 Our Approach

Our approach has three broad phases: (I) Identification of participants, (II) MLN based formulation to identify aliases, and (III) Composite mention creation. We use a *Unified Linguistic Denotation Graph (ULDG)* representation of NLP-processed sentences in the input narrative. The ULDG unifies output from various stages of NLP pipeline such as dependency parsing, NER and coreference resolution, e.g., Figure 1 shows a sample ULDG.

Definition: A ULDG $G(V, E_d, E_p, E_a)$, corresponding to a set S of n sentences, is a vertex-labeled and edge-labeled graph. A node $u \in V$ corresponds to a token in S and its label is defined as: $L_u = (s, t, token, POS, p, a)$; where s : sentence index, t : token index, $token$, POS : part-of-speech tag of $token$, p denotes participant type ($p \in \{PER, ORG, LOC, OTHER\,(OTH)\}$) if u is a headword of a participant mention and a denotes canonical participant mention of corresponding group of aliases. There are three types of edges –

• $E_d = \{\langle u, v, dep\rangle$: directed dependency edge labelled with dep (dependency relation), which connects a governor (parent) token u to its dependent token $v\}$; e.g., ⟨`sent, parent, nsubj`⟩

• $E_p = \{\langle u, v\rangle$: directed edge, which connects headword u of a participant phrase to its each constituent word $v\}$; e.g., ⟨`Bonaparte, Napoleon`⟩

• $E_a = \{\langle u, v\rangle$: undirected edge, which connects nodes u and v which are headwords of aliases of the same participant $\}$; e.g., ⟨`him, Bonaparte`⟩

Our approach has been summarized in Algorithm 1. Its input is an ULDG $G(V, E_d, E_p, E_a)$ for a set S of given sentences. We initialize V, E_d, E_p and E_a using any standard dependency parser, NER and coreference resolution techniques[3].

input : G = ULDG for set of sentences S
output: G with updated participant and alias edges
`// Phase-I: Basic participant mention identification`
foreach $n \in G.nodes$ **do**
  if $n.POS$ *is noun* $\wedge\ n.p = OTH\ \wedge$ $is\_generic\_NP\_head(G, n)$ **then**
    $n.p :=$ $checkWordNetHypernyms(n.token)$
    if $n.p = OTH$ **then** continue
    foreach $\langle n, x, dep\rangle \in E_d$ **do**
      if $dep \in \{$`amod, compound, det`$\}$ **then** $E_p := E_p \cup \{\langle n, x\rangle\}$
foreach $n \in G.nodes$ **do**
  if $n.POS$ *is pronoun* $\wedge\ (\exists x : \langle n, x\rangle \in E_a$ *such that* $x.p \neq OTH)$ **then** $n.p := x.p$
$G := resolveParticipantTypeConflict(G)$
`// Phase-II: MLN-based alias detection`
$E_a := E_a \cup \{\langle u, v\rangle$: where u and v are detected as aliases by *MLN_encoded_Linguistic_Constraints()*$\}$
`// Phase-III: Composite mention creation by merging dependent participant mentions`
$G'(V', E') :=$ Subgraph of G, such that
$V' := \{n \in G : n.p \neq OTH\}$ and
$E' = \{\langle u, v, dep\rangle \in E_d : dep \in \{$`appos, nmod`$\}\}$
foreach $n \in G.nodes$ **do**
  if $n.p = OTH$ **then** continue
  $indParticipant := True$
  foreach $\langle x, n, dep\rangle \in E_d$ **do**
    if $dep \in \{$`appos, nmod`$\} \wedge x.p \neq OTH$ **then** $indParticipant := False$
  if $\neg indParticipant$ **then** continue
  $depParticipants := DFS(G', n)$
  foreach $y \in depParticipants$ **do**
    $E_p := E_p \cup \{\langle n, y\rangle\}$
    foreach $\langle y, x\rangle \in E_p$ **do**
      $E_p := E_p \cup \{\langle n, x\rangle\}$
    $y.p := OTH$
    Drop from E_p all outgoing edges from y
foreach *clique* c *in subgraph* $(V, E_a) \subset G$ **do**
  foreach $n \in c.nodes$ **do**
    n.a := earliest participant mention in $c.nodes$

Algorithm 1: $identify\_participants\_\&\_aliases$

Our algorithm modifies the input ULDG in-place by updating node labels, E_p and E_a. Figure 1 shows an example of initialized input ULDG, which gets transformed by our algorithm to the output ULDG shown in Figure 2.

Phase-I: In this phase, we update participant type

[2] Further details are available in Figure A.1 and Table A.2 in the supplementary material.

[3] We use Stanford CoreNLP Toolkit (Manning et al., 2014)

Predicates	Description
$NEType(x, y)$	y is entity type of participant x
$CopulaConnect(x, y)$	Participants x and y are connected through a copula verb or a "copula-like" verb in E_d (e.g., `become`)
$Conj(x, y)$	Participants x and y are connected by a conjunction in E_d
$DiffVerbConnect(x, y)$	Participants x and y are connected through a "differentiating" verb or a copula-like verb in E_d (e.g. `tell`)
$LexSim(x, y)$	Participants x and y are lexically similar, i.e. having low edit distance
$Alias(x, y)$	Participants x and y are aliases of each other (used as a query predicate)

Hard rules	Description
$Alias(x, x)$; $Alias(x, y) \Rightarrow Alias(y, x)$	Reflexivity and symmetry of aliases
$Alias(x, y) \wedge Alias(y, z) \Rightarrow Alias(x, z)$	Transitivity of aliases
$Alias(x, y) \wedge \neg Alias(y, z) \Rightarrow \neg Alias(x, z)$	
$Alias(x, y) \Rightarrow (NEType(x, z) \Leftrightarrow NEType(y, z))$	If x and y are aliases, their entity types should be same
$Conj(x, y) \Rightarrow \neg Alias(x, y)$	If x and y are conjuncts, then they are less likely to be aliases
Soft rules	**Description**
$CopulaConnect(x, y) \Rightarrow Alias(x, y)$	If x and y are connected though a copula or copula-like verb in E_d , then they are aliases of each other
$LexSim(x, y) \Rightarrow Alias(x, y)$	If x and y are lexically similar, then they are likely to be aliases
$DiffVerbConnect(x, y) \Rightarrow \neg Alias(x, y)$	If x and y are subjects / objects of a "differentiating" verb, then they are not likely to be aliases of each other

Table 2: MLN Predicates and Rules

of headword h of a generic NP if its WordNet hypernyms contain PER/ORG/LOC indicating synsets. We also add new E_p edges from h to dependent nodes of h using dependency relations `compound`, `amod` or `det` (de Marneffe et al., 2014) to get corresponding mention boundaries. The function $resolveParticipantTypeConflict()$ ensures that participant types of all nodes in a single clique in E_a are same by giving higher priority to NER-induced type than WordNet-induced type.

Phase-II: In this phase, we encode linguistic rules in MLN to add new E_a edges. As elaborated by Mojica and Ng (2016), MLN gives the benefits of (i) ability to employ soft constraints, (ii) compact representation, and (iii) ease of specification of domain knowledge.

The predicates and key first-order logic rules are described in Table 2. Here, $Alias(x, y)$ is the only query predicate. Others are evidence predicates, whose observed groundings are specified using G. As we use a combination of hard rules (i.e., rules with infinite weight) and soft rules (i.e., rules with finite weights), probabilistic inference in MLN is necessary to get find most likely groundings of the predicate-$Alias(x, y)$. As the goal is to minimize supervision and to avoid dependence on annotated data, we rely on domain knowledge in the current version to set the MLN rule weights.

Phase-III: In this phase, we extract an auxiliary subgraph $G'(V', E') \subset G$; where V' contains only those nodes which correspond to headwords of basic participant mentions and E' contains only those edges incident on nodes in V' and labeled with `appos` or `nmod`. We identify each independent participant mention in G' and merge its dependent mentions using depth first search (DFS) on G'.

Finally, each clique in E_a represents aliases of an unique participant. We use the earliest non-pronoun mention in text order as the canonical mention for that clique.

4 Experimental Analysis

Datasets: We evaluate our approach on history narratives as they are replete with challenging cases of alias detection. We choose public narratives of varying linguistic complexity to cover a spectrum of history: (i) famous personalities: Napoleon (**Nap**) (Littel, 2008), and Mao Zedong (**Mao**) (Wikipedia, 2018), (ii) a key event: Battle of Haldighati (**BoH**) (Chandra, 2007), and (iii) a major phenomenon: Fascism (**Fas**) (Littel, 2008). We manually annotated these datasets for the independent participant mentions and their aliases. For each alias group of participant mentions we use earliest non-pronoun mention as its canonical mention[4].

We also evaluate it on the newswire subset (**ACE**$_{nw}$) of standard ACE 2005 dataset (Walker et al., 2006). Entity mention annotations were transformed[5] such that only independent entity mentions and their aliases are used. We relied on **Nap** dataset to develop intuition for designing

[4]The annotated datasets are released with this draft.

[5]Transformation scripts are released as supplementary material.

the algorithm and tuning of MLN rules. All other datasets (**ACE**, **BoH**, **Fas**, and **Mao**) are unseen, independent test datasets.

Baselines: B1 is a standard approach to this problem where output of NER and coreference components of Stanford CoreNLP toolkit are combined to detect aliases. B2 is the state-of-the-art coreference resolution system based on (Peng et al., 2015a,b). M is our proposed alias detection approach (Algorithm 1).

Evaluation: The performance of all the approaches is evaluated at two levels: all independent participant mentions (i.e., *participant detection*) and their links with canonical mentions (i.e., *participant linking*). We use the standard F1 metric to measure performance of *participant detection*. For *participant linking*, we evaluate (Pradhan et al., 2014) the combined performance of participant mention identification and alias detection using the standard evaluation metrics, MUC (Vilain et al., 1995), BCUB (Bagga and Baldwin, 1998), Entity-based CEAF (CEAFe) (Luo, 2005) and their average.

Results: Results of the quantitative evaluation are summarized in Table 3. We observe that the proposed approach outperforms other baselines on all datasets.

Dataset & Approach		Participant mentions	Canonical mentions & aliases		
			BCUB	MUC	CEAFe
ACE_{nw}	B1	53.1	38.3	49.4	30.3
	B2	62.9	45.0	50.2	42.5
	M	**70.2**	**52.0**	**56.7**	**50.5**
Nap	B1	60.5	49.4	69.4	32.3
	B2	73.9	56.4	70.2	50.1
	M	**86.4**	**74.1**	**79.0**	**63.6**
BoH	B1	61.7	39.9	56.2	36.2
	B2	65.6	45.0	56.9	40.8
	M	**73.5**	**50.9**	**66.9**	**46.3**
Fas	B1	56.8	40.1	59.3	31.8
	B2	61.6	41.0	54.6	40.3
	M	**70.3**	**55.3**	**64.6**	**51.5**
Mao	B1	60.1	47.4	62.4	38.1
	B2	49.1	29.0	41.9	29.8
	M	**78.9**	**64.1**	**73.9**	**60.2**

Table 3: Experimental results (F_1 metric in %). B1 is combined output of NER and Coreference modules of (Manning et al., 2014). B2 is (Peng et al., 2015a). M is proposed method.

Correct identification of generic NPs as participant mentions, and accurate addition of alias edges due to MLN formulation lead to improved performance of Algorithm 1; e.g., in Table 1, the baselines fail to detect `a lieutenant` as an alias for `Napoleon Bonaparte`, but the proposed approach succeeds as it exploits MLN rule $CopulaConnect(x, y) \Rightarrow Alias(x, y)$. As an illustration of the proposed approach, Table 4 shows the participant mentions and their corresponding canonical mentions for the example text in Table 1.

Sent. no.	Participant Mention	Canonical Mention
1	Napoleon Bonaparte	Napoleon Bonaparte
1	a short man	**Napoleon Bonaparte**
2	he	**Napoleon Bonaparte**
2	his parents	his parents
2	him	**Napoleon Bonaparte**
2	a military school in France	a military school in France
3	he	**Napoleon Bonaparte**
3	a lieutenant	**Napoleon Bonaparte**
4	Napoleon	**Napoleon Bonaparte**
4	the army of the new government	the army of the new government
5	royalist rebels	royalist rebels
5	the National Convention	the National Convention
5	a government official	a government official
5	the young lieutenant	**Napoleon Bonaparte**
5	the delegates	the delegates

Table 4: Output of Algorithm 1 for sentences in Table 1

5 Conclusions

Alias detection is an important and challenging NLP problem. We proposed a linguistically grounded approach to identify aliases of participants in a narrative. We observed that WordNet hypernym tree helps in identification of participant aliases mentioned using generic NPs. MLN proved to be an effective framework to encode linguistic knowledge and achieve better alias detection performance. Our approach was evaluated on history narratives which pose challenging alias detection cases and demonstrated better performance than the state-of-the-art approach. Our goal in current paper was to improve the output by exploiting the strengths (such as linking pronoun mentions to their antecedents) of off-the-shelf coreference algorithms and to overcome their weaknesses (such as resolving generic noun phrase mentions). As part of future work, we are planning to enhance existing MLN frameworks for coreference resolution by integrating the proposed MLN predicates and rules.

References

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*. Granada, volume 1, pages 563–566.

Harsimran Bedi, Sangameshwar Patil, Swapnil Hingmire, and Girish Palshikar. 2017. Event timeline generation from history textbooks. In *Proceedings of the 4th Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA 2017)*. pages 69–77.

Satish Chandra. 2007. *Medieval India: From Sultanat to the Mughals- Mughal Empire: Part Two*. Har Anand Publications.

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford dependencies: A cross-linguistic typology. In *LREC 2014*. pages 4585–4592.

Claire Gardent, Hélène Manuélian, and Eric Kow. 2003. Which bridges for bridging definite descriptions? In *Proceedings of 4th International Workshop on Linguistically Interpreted Corpora (LINC-03) at EACL 2003*.

Paul Hsiung, Andrew Moore, Daniel Neill, and Jeff Schneider. 2005. Alias detection in link data sets. In *Proceedings of the International Conference on Intelligence Analysis*. volume 4.

McDougal Littel. 2008. *World History: Patterns of Interaction*. World History: Patterns of Int. Houghton Mifflin Harcourt Publishing Company.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *HLT/EMNLP 2005*. Association for Computational Linguistics, pages 25–32.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *ACL 2014*. pages 55–60.

Luis Gerardo Mojica and Vincent Ng. 2016. Markov logic networks for text mining: A qualitative and empirical comparison with integer linear programming. In *LREC 2016*.

Vincent Ng. 2017. Machine learning for entity coreference resolution: A retrospective look at two decades of research. In *AAAI, 2017*. pages 4877–4884.

Haoruo Peng, Kai-Wei Chang, and Dan Roth. 2015a. A Joint Framework for Coreference Resolution and Mention Head Detection. In *CoNLL 2015*. pages 12–21.

Haoruo Peng, Daniel Khashabi, and Dan Roth. 2015b. Solving Hard Coreference Problems. In *NAACL HLT 2015*. pages 809–819.

Massimo Poesio, Renata Vieira, and Simone Teufel. 1997. Resolving bridging references in unrestricted text. In *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*. ANARESOLUTION '97, pages 1–6.

Hoifung Poon and Pedro M. Domingos. 2008. Joint Unsupervised Coreference Resolution with Markov Logic. In *EMNLP 2008*.

Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*. volume 2014, pages 30–35.

Matthew Richardson and Pedro M. Domingos. 2006. Markov logic networks. *Machine Learning* 62(1-2):107–136.

Emili Sapena, Lluís Padró, and Jordi Turmo. 2007. Alias Assignment in Information Extraction. *Procesamiento del Lenguaje Natural* 39.

Yang Song, Jing Jiang, Wayne Xin Zhao, Sujian Li, and Houfeng Wang. 2012. Joint learning for coreference resolution with markov logic. In *EMNLP-CoNLL 2012*. pages 1245–1254.

Renata Vieira and Simone Teufel. 1997. Towards resolution of bridging descriptions. In *ACL-EACL 1997*. pages 522–524.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*. Association for Computational Linguistics, pages 45–52.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia* 57.

Wikipedia. 2018. Mao zedong — Wikipedia, the free encyclopedia. [Online; accessed 22-Feb-2018]. https://en.wikipedia.org/wiki/Mao_Zedong.