A New Approach to Animacy Detection

Labiba Jahan[1], **Geeticka Chauhan**[2], **Mark A. Finlayson**[1]
[1]School of Computing and Information Sciences
Florida International University, Miami, FL 33199
[2]Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology, Cambridge, MA
{ljaha002, markaf}@fiu.edu, geeticka@mit.edu

Abstract

Animacy is a necessary property for a referent to be an agent, and thus animacy detection is useful for a variety of natural language processing tasks, including word sense disambiguation, co-reference resolution, semantic role labeling, and others. Prior work treated animacy as a word-level property, and has developed statistical classifiers to classify words as either animate or inanimate. We discuss why this approach to the problem is ill-posed, and present a new approach based on classifying the animacy of co-reference chains. We show that simple voting approaches to inferring the animacy of a chain from its constituent words perform relatively poorly, and then present a hybrid system merging supervised machine learning (ML) and a small number of hand-built rules to compute the animacy of referring expressions and co-reference chains. This method achieves state of the art performance. The supervised ML component leverages features such as word embeddings over referring expressions, parts of speech, and grammatical and semantic roles. The rules take into consideration parts of speech and the hypernymy structure encoded in WordNet. The system achieves an F_1 of 0.88 for classifying the animacy of referring expressions, which is comparable to state of the art results for classifying the animacy of words, and achieves an F_1 of 0.75 for classifying the animacy of coreference chains themselves. We release our training and test dataset, which includes 142 texts (all narratives) comprising 156,154 words, 34,698 referring expressions, and 10,941 co-reference chains. We test the method on a subset of the OntoNotes dataset, showing using manual sampling that animacy classification is 90%$\pm$2% accurate for coreference chains, and 92%$\pm$1% for referring expressions. The data also contains 46 folktales, which present an interesting challenge because they often involve characters who are members of traditionally inanimate classes (e.g., stoves that walk, trees that talk). We show that our system is able to detect the animacy of these unusual referents with an F_1 of 0.95.

1 Introduction

Animacy is the characteristic of being able to independently carry out actions (e.g., movement, communication, etc.). For example, a person or a bird is animate because they move or communicate under their own power. On the other hand, a chair or a book is inanimate because they do not perform any kind of independent action.

Animacy is a useful semantic property for different NLP systems, including word sense disambiguation (WSD), semantic role labeling (SRL), coreference resolution, among many others. Animacy can be used to distinguish different senses and thus help a WSD systems assign senses to different words. As an example, animacy has been applied in grouping senses from WordNet (Palmer et al., 2004; Palmer et al., 2007). Animacy can also be used directly in a WSD system to decide thematic assignment, which is useful for assigning senses: for example, Carlson and Tanenhaus (1988) used the presence of an animate subject in a sentence to determine if a the verb is transitive, which is a useful for thematic role assignment. Another task where animacy can play an important role is semantic role labeling (SRL). Agentive

This work is licensed under a Creative Commons Attribution 4.0 International License. License details: http://creativecommons.org/licenses/by/4.0/

Proceedings of the 27th International Conference on Computational Linguistics, pages 1–12
Santa Fe, New Mexico, USA, August 20-26, 2018.

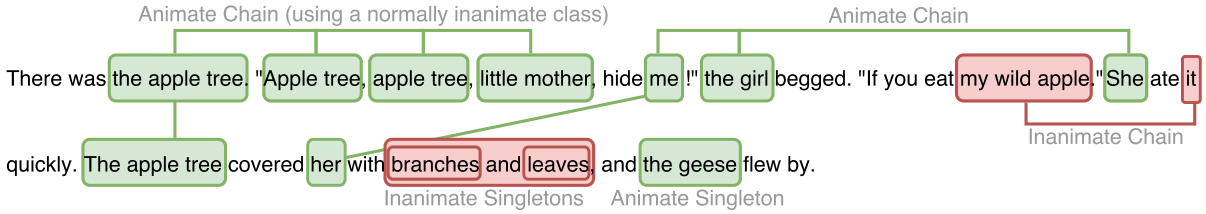

Figure 1: Example text containing animate and inanimate coreference chains. Colored boxes represent referring expressions, while links between them signify coreference. Animate chains are green, while inanimate chains are red. The text is drawn from Story #113 *The Magic Swan Geese* (Guterman, 1975, p. 350) and has been slightly modified for clarity. The figure is adapted from (Jahan et al., 2017).

or semantic subject roles must often be filled by animate entities, whereas goal, theme, patient, instrument and location roles are often filled by inanimate entities (Kittila et al., 2011). In some works (Connor et al., 2013; Kittilä, 2006, for example), animacy is used as a feature that helps to identify agents, and Ferreira (1994) showed how knowing the animacy of roles allows one to better identify the passive voice. In many coreference resolution systems (Raghunathan et al., 2010; Iida et al., 2003; Cardie and Wagstaf, 1999, for example) animacy is used as an semantic feature to determine co-referents of an expression.

In addition to these broad uses of animacy, our own research group is particularly interested in detecting animacy with a view toward identifying characters in stories. Most definitions of narrative acknowledge the central role of character, for example: "a representation of a possible world . . . at whose centre there are one or several protagonists of an *anthropomorphic* nature . . . who (mostly) perform goal-directed actions . . . " *(emphasis ours)* (Fludernik, 2009, p. 6). If we are to achieve the long-term goal of automatic story understanding, it is critical that we be able to automatically identify a story's characters, distinguishing them from non-character entities. All characters are necessarily animate—although not all animate things are necessarily characters—and so detecting animacy will immediately narrow the set of possibilities for character detection.

Prior work treated animacy as a word-level phenomenon, marking animacy as an independent feature on individual words (Orăsan and Evans, 2007; Bowman and Chopra, 2012; Karsdorp et al., 2015). But word-level animacy is not always sufficient to identify an animate or an inanimate object. For example, *horse* is normally animate, but a *dead horse* is obviously inanimate. On the other hand, *tree* is an inanimate word but a *talking tree* is definitely an animate thing. So, assigning animacy at the word level confuses the issue and makes it more difficult classify these type of complex cases.

Furthermore, referents are expressed in texts as coreference chains comprised of referring expressions, and so conceiving of animacy as a word-level phenomenon requires an additional method for computing chain animacy from word animacy. One way to do this is to combine word-level animacy markings—say, using majority vote—into referring expressions animacy and then coreference chains. As it turns out, this does not work all that well and we used this method as our baseline. Alternatively, we can attempt to compute animacy directly on the referring expressions and then use majority vote of referring-expression-level animacy to compute animacy of coreference chains, the approach we pursue here.

Although detecting animacy might seem to be straightforward, it presents a number of subtleties. For example, some theorists have proposed closed lists of linguistic expressions that should be automatically considered animate entities, such as titles, animals, or personal pronouns (Quirk et al., 1985; Yamamoto, 1999). However, texts, especially stories about unreal worlds, can arbitrarily introduce characters that would not be animate in real life, for example, walking stoves or talking trees. Figure 1 shows an example sentence from a Russian fairytale which contains three animate chains, one of which is a tree that talks: trees would not be normally be considered animate according to canonical lists of animate entities. Therefore some context sensitivity in detection is needed.

In our work, we compute animacy directly on referring expressions, and transfer those markings up to the coreference chain level, to get a direct classification of the animacy of the whole chain. We present a hybrid system combining statistical machine learning (ML) and hand-built rules for classifying the

Text Types	# Texts	# Tokens	# Referring Expressions	# Coref. Chains
Russian Folktales	46	109,120	20,391	4,950
Islamist Extremist Texts	32	26,557	8,041	3,684
Islamic Hadiths	64	20,477	6,266	2,307

Table 1: Counts of various text types in the corpus.

animacy of referring expression, and also present a voting model to identify the animacy of coreference chains based on the animacy of the chain's constituent referring expressions. The paper proceeds as follows. First we discuss our data sources and annotation procedures (§2). Next we discuss the experimental setup including the ML features, rules, and classification models (§3), and then describe our results (§4). We analyze the error patterns of the system and mention potential future work (§5), and also discuss work that is related to this study (§6). We finish with a discussion of the contributions of the paper (§7).

2 Data

We started this project seeking to use existing data annotated for animacy, as there have been a number of studies on animacy detection already (as we discuss in §6). However, no prior data in English was readily available to use; the best performing prior work on word-level animacy was done on a corpus of 74 stories comprising 74,504 words in Dutch (Karsdorp et al., 2015). Orăsan and Evans (2007) did their work in English but their data was not available. Therefore we sought other data (specifically stories, because of our interest in story understanding), and our annotated data was a corpus comprising a variety of Russian folktales, Islamist Extremist stories, and Islamic Hadiths that are freely available and assembled for other work, and had been annotated for referring expressions and coreference chains (Finlayson, 2017; Finlayson et al., 2014). The composition of the corpus is shown in Table 1.

The corpus contains 46 Russian folktales, originally collected in Russian in the late 1800's but translated into English in the mid-twentieth century (Finlayson, 2017). The other portion (the N2 corpus) contains 96 stories of relevance to Islamist Extremists (Finlayson et al., 2014). All but 31 of the texts in the corpus already contained gold-standard annotations for token and sentence boundaries, parts of speech, referring expressions, and coreference chains (as well as other layers of annotation. We processed these 31 un-annotated texts using the Stanford CoreNLP suite (Manning et al., 2014), automatically generating tokens, sentences, parts of speech, referring expressions, and coreference chains.

We annotated the whole corpus for animacy of coreference chains, and the first fifteen stories for animacy at the word level. We propagated the animacy annotations from the chains to their constituent referring expressions to generate animacy annotations at that level. Because we had to automatically compute referring expression and coreference chains on 31 of the texts, and the CoreNLP coreference resolution is somewhat noisy, we hand-corrected the chains. We did this hand-correction using the Story Workbench annotation tool (Finlayson, 2008; Finlayson, 2011) that allows for the manipulation and correction of referring expression and coreference chains.

The annotation of the animacy of coreference chains and referring expressions for the first fifteen stories was performed by the first two co-authors. Disagreements were discussed and corrected to generate a gold-standard annotation. Agreement for the coreference-level was 0.99 F_1 and 0.99 Cohen's kappa coefficient (κ), which represents near-perfect overall agreement (Landis and Koch, 1977). The annotation of the rest of the stories was performed by only the first author.

We also annotated first fifteen Russian tales for word-level animacy so that we could test via reimplementation the existing best performing word animacy model (Karsdorp et al., 2015). This annotation was done under the following guidelines. First, all nouns that would refer to animate entities in real life, such humans or animals, as discussed in (Quirk et al., 1985, pp. 314 & 345) were marked animate. We marked gendered pronouns as animate, e.g., *he*, *she*, *his*, *hers*, etc. We also marked adjectives suggesting animacy as animate, e.g., *alive*, *vital*, *kindlier*, etc., whereas adjectives implying inanimacy, such as

	Total entity	Animate entity	Inanimate entity	Unique Animate	Unique Inanimate
Token (15 stories)	23,291	3,896	19,395	291	2,221
Referring Expression (142 stories)	34,698	22,052	12,646	1,104	2,249
Coreference-chain (142 stories)	10,941	3,832	7,109	-	-

Table 2: Total number of animate and inanimate tokens, referring expressions, and coreference chains, with breakdowns of number of unique items in each class.

Referring Expression	**Class**	**Explanation**
Muslims, the dragon, Abu Bakr	Animate	Normally animate entities
walking stove, talking tree	Animate	Normally inanimate but are animate in context
"those who do not know what it is"	Inanimate	Discourse acts, when marked as referents
the mosque, this world, every house	Inanimate	Normally inanimate objects
dead horse	Inanimate	Normally animate but are inanimate in context
her eyes, his hands , horse tail	Inanimate	Inanimate parts of animate entities
Word		
princess, dragon, Abdullah	Animate	Nouns denoting animate entities
he, she, his, her	Animate	Personal pronouns referring to animate objects
kind [prophet], stronger [dragon]	Animate	Adjectives that suggest animacy
Morning, Evening, [talking] stove	Animate	Usually inanimate but are animate in context
Kiev, world, mosque	Inanimate	Nouns denoting inanimate entities
it, that, this	Inanimate	Personal pronouns referring to inanimate objects

Table 3: Examples of annotation of coreference- and word-level animacy. At the word level, only an adjectives suggesting animacy or nouns referring to an animate object are marked animate. Everything else (including verbs, adverbs, determiners, and so forth) are marked inanimate.

dead in the noun phrase *dead horse*, were marked inanimate. Second, we marked as animate any words directly referring to entities that acted animately in a story, regardless of the default inanimacy of the words. For example, we marked *stove* animate in the case of a walking stove, or *tree* animate in the case of a talking tree. This also covered proper names that might normally be marked as inanimate because of their ostensible class, such as those underlined in the next example:

> *All of them were born in one night—the eldest in the evening, the second at midnight, and the youngest in the early dawn, and therefore they were called Evening, Midnight, and Dawn. (Guterman, 1975, Tale #140, p. 458)*

The word-level annotation was done by the first two co-authors. Disagreements were discussed and corrected to generate a gold-standard annotation. We annotated every word in the corpus for animacy directly (marking each word as either animate or not). Agreement was 0.97 F_1 and 0.97 Cohen's kappa coefficient (κ), which represents near-perfect overall agreement (Landis and Koch, 1977).

A summary of the counts of animate and inanimate words, referring expressions, and coreference chains is given in Table 2. Examples of animate and inanimate words are given in Table 3. The data is included in the supplementary materials archive for the paper, which is publicly available[1].

3 Approach

Our hybrid system comprises two parts: a rule-based classifier that can mark the animacy of roughly 50% of the referring expressions, followed by a statistical classifier trained on the annotated data that can be

[1] https://dspace.mit.edu/handle/1721.1/116172

applied to the remaining referring expressions. Once all referring expressions are marked for animacy, the animacy of a coreference chain is inferred from the animacy of its constituent referring expression.

3.1 Rules

We implemented five rules that considered semantic subjects parsed from the semantic role labeler associated with the Story Workbench annotation tool (Finlayson, 2008; Finlayson, 2011), the named entities computed using the classic API of Stanford dependency parse (Manning et al., 2014, v3.7.0), and knowledge from WordNet (Fellbaum, 1998). These rules were inspired by existing rule-based animacy systems. We also considered the last word of a referring expression in most of the rules because it helps to mark quotes as inanimate, as well as to detect the regular animate and inanimate referring expression.

1. If the last word of a referring expression is a gendered personal, reflexive, or possessive pronoun (i.e., excluding *it*, *its*, *itself*, etc.), we marked it animate.
2. If the last word of a referring expression is the semantic subject to a verb, we marked it animate.
3. If a referring expression contains a proper noun we marked it animate. We excluded anything tagged as *location*, *organization*, or *money*, as determined by the Stanford CoreNLP NER system.
4. If the last word of a referring expression is a descendant of *living_being* in WordNet, we marked it animate.
5. If the last word of a referring expression is a descendant of *entity* WordNet, we marked it inanimate.

3.2 Features

We explored seven different binary and vector features to train the statistical classification model, some of which are drawn from prior work.

1. **Word Embeddings (WE)**: We computed pre-trained word embeddings in 300 dimensions for all the words in the stories using the skip-gram architecture algorithm (Mikolov et al., 2013). We used the DeepLearning4J library (Deeplearning4j Development Team, 2017), and configured the built-in skip-gram model with a minimum word frequency of 3, layer width (dimensions) of 300, a window size of 5, and trained for 10 iterations. We explored a few different combinations of these parameters, but found that these settings produced the best results. This is a vector feature drawn from (Karsdorp et al., 2015), and is primarily relevant to classifying word-level animacy. We ran this model on each word of our data and used the output vector as a feature.

2. **Word Embeddings on Referring Expressions (WER)**: We calculated pre-trained word embeddings in 450 dimensions for just the words within the referring expressions, again using the skip-gram approach as above, except with a minimum word frequency of 1. Again, this is a vector feature. 450 dimensions worked better for this feature (rather than 300), which we discovered after doing a small amount of parameter exploration. We ran this model on each referring expression of our data used the output vector as a feature.

3. **Composite Word Embedding (CWE)**: We computed a composite pre-trained word embedding for the neighborhood of each word, adding together the word embedding vectors for three words before and three words after the target word (excluding the target). This is also a vector feature, and is again partially drawn from (Karsdorp et al., 2015). The idea of this feature is that it estimates the similarities of the context among all animate words (or all inanimate words) as well as the dissimilarities of animate from inanimate, and vice versa.

4. **Parts of Speech (POS)**: By analogy with the other embeddings, we computed an embedding over part of speech tags in 300 dimensions, with the same settings as in feature #1 (WE). This feature models the tendency of nouns, pronouns, and adjectives to refer to animate entities.

5. **Noun (N)**: We checked whether a given referring expression contained a noun and encoded this as a boolean feature because we observed that in the first 15 stories 43% of nouns are animate. Thus this feature explicitly captures the tendency of nouns to refer to animate entities. We used dependency parses generated by the classic API of Stanford dependency parser (Manning et al., 2014, v3.7.0).

6. **Grammatical Subject (GS)**: Animate references tend to appear as the grammatical subjects of verbs (Ovrelid, 2005). We used dependency parses generated by the classic API of Stanford dependency parser (Manning et al., 2014, v3.7.0) to check if the last word of a given referring expression was used as a grammatical subject relative to any verb in the sentence, and encoded this as a boolean feature.

7. **Semantic Subject (SS)**: We also computed whether or not a referring expression appeared as a semantic subject to a verb. We used the semantic role labeler associated with the Story Workbench annotation tool (Finlayson, 2008; Finlayson, 2011) to compute semantic roles for all the verbs in the stories. We then checked whether the last word of a given referring expression contained an ARG0 for a verb (an exact match was not required), and encoded this as a boolean feature.

3.3 Classification Models

We implemented our classification models using SVM (Chang and Lin, 2011), with a Radial Basis Function Kernel. The features used to train the different models are shown in Table 4. We trained each model using cross validation, and report macroaverages across the performance on test folds.

We have three models for animacy: referring expressions, coreference chains, and words. For our referring expression animacy model, we implemented two approaches. The first is a ML-only approach, in which we explored different combinations of features: word embedding over referring expressions (WER), noun (N), grammatical subject (GS), and semantic subject (SS). We configured the SVM with $\gamma = 1, C = 0.5$ and $p = 1$. We measured the performance of the classifier using 10-fold cross validation. The second approach is the hybrid system where we we first applied the rules, then applied the ML classifier for referring expressions not covered by the rules. In our prior work we only implemented the first approach (Jahan et al., 2017) on a small data set.

For the coreference chain animacy model, we implemented a majority voting approach for combining the results of the referring expression animacy model to obtain a coreference animacy prediction. In the case of ties, the chain was marked inanimate.

To compare with prior work, we also implemented a word animacy model, adapting an existing system with the best performance (Karsdorp et al., 2015). That model used features based on word N-grams, parts of speech, and word embeddings. Similarly, we implemented our classifier using word embeddings over words (WE), combined word embeddings (CWE), and parts of speech (POS). The SVM was configured with $\gamma = 5, C = 5000$ and $p = 1$, and we measured the performance with 20-fold cross validation. This model performed very close to the prior state of the art with our small data set of 15 stories.

4 Results & Discussion

We calculated two baselines for referring expression animacy. The first baseline is to choose the majority class (animate). The second baseline combines word-level animacy predictions generated by our word animacy model via a majority vote; we measured the upper bound for this over the 15 texts for which we have gold-standard word animacy annotations.

We evaluated our models by measuring accuracy, precision, recall, F_1, and Cohen's kappa (κ) compared to the gold-standard annotations. Table 4 shows the results for both classes. Our word animacy model achieved an F_1 of 0.98, whereas the prior state of the art achieved F_1 of 0.99 for marking inanimacy. On the other hand, for marking animacy our model achieved F_1 of 0.90 where the state of the art achieved F_1 of 0.93. For referring expression animacy we varied the features to determine the optimal set. We obtained the best result (F_1 of 0.84) using different combinations of three features: noun (N), grammatical subject (GS) and semantic subject (SS). Our hybrid model for referring expression animacy performed better (F_1 of 0.88) than the statistical model (F_1 of 0.84). The rule-based model achieved 0.88 F_1 when we applied the rules first, and marked any remaining referring expressions as majority class. Our rule based model performed similarly to the hybrid model, but the hybrid model is more consistent.

For the coreference animacy model, we implemented the majority vote approach to detect animacy of coreference chain using the best output of referring expression model. Majority vote resulted in an overall F_1 of 0.75, which substantially outperforms the result from our prior work of 0.61 F_1. Around 3% of coreference chains resulted in a tied vote, and these were marked as inanimate (the majority class).

Model	Feature Set	Acc.	Inanimate				Animate			
			κ	Prec.	Rec.	$\mathbf{F_1}$	κ	Prec.	Rec.	$\mathbf{F_1}$
Word	(Karsdorp et al., 2015)	-	-	0.98	0.99	0.99	-	0.94	0.91	0.93
	WE, CWE, POS	96%	0.87	0.98	0.98	0.98	0.87	0.91	0.88	0.90
Ref. Expr.	Baseline MFC	61%	0.0	0.0	0.0	0.0	0.0	0.61	1.0	0.76
	Baseline Maj. Vot.	75%	0.53	0.59	0.99	0.74	0.53	0.99	0.62	0.76
	Hybrid on Stanford	80%	0.61	0.89	0.73	0.79	0.61	0.74	0.90	0.81
	WER, N, GS, SS	76%	0.47	0.80	0.51	0.62	0.47	0.76	0.92	0.83
	N, GS	78%	0.51	0.83	0.54	0.65	0.51	0.77	0.93	0.84
	N, SS	79%	0.53	0.80	0.60	0.68	0.53	0.78	0.91	0.84
	N, GS, SS	79%	0.53	0.81	0.59	0.68	0.53	0.78	0.91	0.84
	Prior best result	86%	0.70	0.83	0.77	0.80	0.68	0.87	0.91	0.90
	Rule Based model	82%	0.60	0.89	0.60	0.72	0.60	0.81	0.96	0.88
	Hybrid model	83%	0.62	0.84	0.67	0.74	0.62	0.83	0.93	0.88
	Random Sampling	92%*	0.85	0.87	0.93	0.91	**0.85**	**0.96**	**0.92**	**0.94**
Coref.	Prior best result	79%	0.48	0.93	0.80	0.86	0.48	0.50	0.76	0.61
	Maj. vote on Stanford	79%	0.54	0.91	0.78	0.84	0.54	0.60	0.82	0.69
	Maj. vote (current)	82%	0.61	0.87	0.84	0.86	0.61	0.73	0.77	0.75
	Random Sampling	90%†	0.80	0.86	0.98	0.92	**0.80**	**0.97**	**0.81**	**0.88**

Table 4: Result of different Animacy Models (Bolded according to when our F_1 measure is higher). MFC stands for "Most Frequent Class", and the other abbreviations stand for features as indicated in the text. *Estimated $\pm 2\%$ with 95% confidence. †Estimated $\pm 1\%$ with 95% confidence.

We also evaluated our model using direct sampling (Saunders et al., 2009). We ran our hybrid model over 200 news articles from the OntoNotes (Hovy et al., 2006) data set containing 46,088 referring expressions and 7,836 coreference chains. We randomly sampled 558 coreference chains and checked their animacy markings by hand, resulting in a estimated accuracy of 90% $\pm 2\%$ at a 95% confidence level, as well as estimated precision, recall, and F_1 listed in Table 4. Those coreference chains contained 3,543 referring expressions, which allowed us to estimate the accuracy of the referring expression model at 92% $\pm 1\%$ at a 95% confidence level.

The data contains 46 folktales, which have 142 mentions of 12 characters who are members of traditionally inanimate classes (e.g., stoves that walk, trees that talk). We manually identified those 12 characters and evaluated our model's performance on them. Our system is able to detect the animacy of these unusual referents with an F_1 of 0.95. Conversely, there was only one mention of a normally animate class that was inanimate in context ("dead horse"), and this was correctly marked by the system.

5 Error Analysis & Future Work

A detailed error analysis of the results revealed at least two major problems for the hybrid model that we will focus on in future work: short chains, quotes, and exceptions to the rules.

Determining the animacy of short coreference chains was challenging for our system: approximately 11% of short chains are incorrectly marked. As the length of a chain tends toward a single referring expression, the coreference classifier should converge to the referring expression classifier performance. However, for chains between two and four referring expressions long, the majority voting approach seems to fall short. We suspect this is because many referring expressions are themselves quite short, and can contain false alarms: e.g., our system classifies "his hands" as animate because of the animate word "his" in the expression. We believe another approach to solving this problem is to generate new rules in our hybrid model so that it can handle these type of special cases.

Second, many quotes are full of animate words, e.g., "the fate of the tsar's daughter to go to the dragon" is a phrase that is itself a referring expression in one story, and should be inanimate according

to our animacy annotation rule. However, the classifier marks the quote as animate because it finds three animate words: *tsar*, *daughter*, and *dragon*. In our data, approximately 2.5% of quotes that are referring expressions are incorrectly marked, and handling this likely will require rule-based processing.

Finally, a common error type was exceptions to the rules. In the hybrid system we combined together a large number of similar referring expressions under one rule so that we can handle them under a similar animacy class. But there are always exceptions for every rule: for example, we define "it" as inanimate but of course sometimes "it" can refer to an animate object. For the most part these individual instances will be out-voted by animate referring expressions in long chains, so it is a relatively small problem. One approach to solving this would to implement the idea of Orăsan and Evans (2001; 2007) to use supervised machine learning to mark unseen WordNet senses by their animacy rather using specific rules.

6 Related Work

We divide the related work into two sections: first animacy detection in English, followed by animacy detection in other languages. The work reported here is in English (thus the related work of the first section), but the material covered in non-English second section makes clear both that our approach had not attempt before in any language, and also that no language-specific features have been used in any prior work. There have been both rule-based and machine learning methods to classify the animacy of words, but to the best of our knowledge, no one has combined both techniques, and no one has tackled animacy classification at the referring expression or coreference level.

6.1 Animacy Detection in English

Evans and Orăsan (2000) performed animacy classification to improve anaphora resolution using a rule-based method to identify animate WordNet hypernym branches. In later work they used supervised machine learning to mark unseen WordNet senses for their animacy (Orăsan and Evans, 2001; Orăsan and Evans, 2007). The rule-based method uses the unique beginners in WordNet for classification of sense animacy using a statistical chi-squared method, while the machine learning method uses k-nearest neighbors in a multi-step procedure, along with careful feature engineering, to determine noun animacy. They achieved an F_1 of 0.94 for animacy, and also performed an extrinsic evaluation using the MARS anaphora resolution system and a word sense disambiguation algorithm. Similarly, Moore et al. (2013) combined a majority vote model using rule-based methods, features from WordNet, and a SVM to achieve an accuracy of 89% for majority voting and 95% for SVM (no F_1 score was reported).

Bowman and Chopra (2012) used a maximum entropy classifier to predict multiple classes for noun phrases as *human*, *vehicle*, *time*, *animal*, etc., with an overall accuracy of 85%. A binary animacy classification could be derived from each of these classes, with a performance of 94% accuracy.

Additionally, there are others that have used pure rule-based and pattern matching methods. Ji and Lin (2009) generate n-grams and performed pattern matching using the Google n-gram corpus to label gender and animacy properties for words for to assist in person mention detection. With these gender and animacy markings, they applied a confidence estimation which is compared against the test document using fuzzy matching. The highest F_1 they achieved for animacy was 0.67, with an F_1 of 0.46 for gender.

Declerck et al. (2012) used an ontology-based method to detect characters in folktales. Their ontology consists of family relations as well as elements of folktales such as supernatural entities. After looking at the heads of noun phrases and comparing them with labels in the ontology, they added the noun phrase to the ontology as a potential character if a match was found. Then, they applied inference rules to the candidate characters in order to find two strings in the text that refer to the same character. They discarded strings that are related to a potential character only once and are not involved in an action. They obtained an accuracy of 79%, a precision of 0.88, a recall 0.73, and an F_1 of 0.80.

Wiseman et al. (2015) used a mention-ranking approach for coreference resolution, using animacy as a feature, derived from the Stanford Coreference System (Lee et al., 2013). The Stanford Coreference System set animacy attributes using a static list for pronouns, named entity labels, and a dictionary.

Finally, a marginally related rule-based system was implemented by Goh et al. (2012) using verbs and WordNet in order to determine the protagonists in fairy tales (where protagonists must of necessity be

animate). They used the Stanford parser's phrase structure trees to obtain the subjects and objects of the verbs and used the dependency structure to obtain the head noun of compound phrases. Additionally, they used WordNet's *derivationally_related* relation to find verb associated with a particular nominal action. They achieved a precision of 0.69, a recall of 0.75, and an F_1 of 0.67.

6.2 Animacy Detection in Other Languages

Nøklestad (2009) implemented animacy detection for Norwegian nouns, using this along with Named Entity Recognition to improve the performance of anaphora resolution. They explored various pattern matching methods, using web data to extract lists of animate nouns as well as to check the animacy of a particular noun. For example, if a noun co-referred frequently with *han* (he) or *hun* (she), then it was characterized as animate. This method achieved an accuracy of 93%. The main problem here, from our point of view, is that using data from the web makes the problem too general: you only measure the typicality of animacy, not the animacy of an item in context. In the case of folktales, we have unusual animate entities (e.g., talking stoves) that will on the whole be seen by the web as inanimate.

Bloem and Bouma (2013) developed an automatic animacy classifier for Dutch nouns by dividing them into *Human*, *Nonhuman* and *Inanimate* classes. They use the k-nearest neighbor algorithm with distributional lexical features—e.g., how frequently the noun occurs as a subject of the verb "to think" in a corpus—to decide whether the noun was predominantly animate. Prediction of the *Human* category achieved 87% accuracy, and the large inanimate class was predicted correctly 98% of the time. But, again, this work focuses on individual noun phrases, not coreference chains, and is concerned with the default animacy of the expression, not its animacy in context.

Another implementation of word-level animacy for Dutch was performed by Karsdorp et al. (2015) on folktale texts. Because this work was the highest performing word-level system, many of our features were inspired by their approach. They used lexical features (word forms and lemmas), syntactic features (dependency parses to check which word is a subject or an object), part of speech tags, and semantic features (word embedding using a skip-gram model to vectorize each word). They implemented a Maximum Entropy Classifier to classify words according to their animacy and obtained a good result of 0.93 F_1 for the animate class, by just using the words, parts of speech, and embedding features.

Baker and Brew (2010) performed animacy classification on a multilingual dataset containing English and Japanese. They used Bayesian logistic regression with morphological features, WordNet semantic categories, and frequency counts of verb-argument relations. They obtained 95% classification accuracy. In sum, all the prior work has been for word-level animacy (usually nouns, sometimes noun phrases). In contrast, we focus on characterizing the animacy of referring expressions and coreference chains.

7 Contributions

This paper makes four major contributions. First, we have redefined the problem of animacy classification as one of marking animacy on coreference chains, in contrast to all prior work that seeks to mark the animacy at the world level. Second, we have presented a hybrid system merging an SVM classifier and hand-built rules to predict the animacy of referring expressions directly, achieving performance of 0.90 F_1, which is comparable to the state of the art for word-level animacy detection. Third, we used a majority voting approach to obtain the animacy of coreference chains. The overall performance of this approach is substantially improved in comparison with our prior work. Our error analysis further suggests several potentially profitable ways forward to improving the performance. Finally, we provide 15 texts annotated for word-level animacy and 142 texts annotated for coreference chain animacy, as well as the code reproducing the results, in the supplementary materials archive[2].

Acknowledgements

This work was partially supported by Office of Naval Research (ONR) grant number N00014-17-1-2983. We would also like to thank Deya Banisakher, Joshua Eisenberg and W. Victor H. Yarlott from the FIU Cognac Lab for their discussions and assistance.

[2] https://dspace.mit.edu/handle/1721.1/116172

References

Kirk Baker and Chris Brew. 2010. Multilingual animacy classification by sparse logistic regression. *Information Concerning OSDL OHIO STATE DISSERTATIONS IN LINGUISTICS*, page 52.

Jelke Bloem and Gosse Bouma. 2013. Automatic animacy classification for Dutch. *Computational Linguistics in the Netherlands Journal (CLIN)*, 3:82–102.

Samuel R Bowman and Harshit Chopra. 2012. Automatic animacy classification. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT): Student Research Workshop*, pages 7–10. Montreal, Canada.

Claire Cardie and Kiri Wagstaf. 1999. Noun phrase coreference as clustering. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

Greg N Carlson and Michael K Tanenhaus. 1988. Thematic roles and language comprehension. *Syntax and semantics*, 21:263–288.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Michael Connor, Cynthia Fisher, and Dan Roth. 2013. Starting from scratch in semantic role labeling: Early indirect supervision. In *Cognitive aspects of computational language acquisition*, pages 257–296. Springer.

Thierry Declerck, Nikolina Koleva, and Hans-Ulrich Krieger. 2012. Ontology-based incremental annotation of characters in folktales. In *Proceedings of the 6th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 30–34. Association for Computational Linguistics.

Deeplearning4j Development Team. 2017. Deeplearning4j: Open-source distributed deep learning for the JVM, Apache Software Foundation License 2.0. `http://deeplearning4j.org`. Accessed: 2017-04-08.

Richard Evans and Constantin Orăsan. 2000. Improving anaphora resolution by identifying animate entities in texts. In *Proceedings of the Discourse Anaphora and Reference Resolution Conference (DAARC2000)*, pages 154–162. Lancaster, England.

Christiane Fellbaum, editor. 1998. *WordNet An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Fernanda Ferreira. 1994. Choice of passive voice is affected by verb type and animacy. *Journal of Memory and Language*, 33(6):715–736.

Mark A Finlayson, Jeffry R Halverson, and Steven R Corman. 2014. The n2 corpus: A semantically annotated collection of islamist extremist stories. In *LREC*, pages 896–902.

Mark Alan Finlayson. 2008. Collecting semantics in the wild: The story workbench. In *Proceedings of the AAAI Fall Symposium on Naturally Inspired Artificial Intelligence*, pages 46–53. Arlington, VA.

Mark A Finlayson. 2011. The Story Workbench: An extensible semi-automatic text annotation tool. In *Proceedings of the 4th Workshop on Intelligent Narrative Technologies (INT4)*, pages 21–24. Stanford, CA.

Mark A. Finlayson. 2017. ProppLearner: Deeply Annotating a Corpus of Russian Folktales to Enable the Machine Learning of a Russian Formalist Theory. *Digital Scholarship in the Humanities*, 32(2):284–300.

Monika Fludernik. 2009. *An Introduction to Narratology*. Routledge, New York.

Hui-Ngo Goh, Lay-Ki Soon, and Su-Cheng Haw. 2012. Automatic identification of protagonist in fairy tales using verb. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 395–406. Springer.

Norbert Guterman. 1975. *Russian Fairy Tales*. Pantheon Books, New York.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: the 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.

Ryu Iida, Kentaro Inui, Hiroya Takamura, and Yuji Matsumoto. 2003. Incorporating contextual cues in trainable models for coreference resolution. In *Proceedings of the EACL Workshop on The Computational Treatment of Anaphora*, pages 23–30. Citeseer.

Labiba Jahan, Geeticka Chauhan, and Mark Finlayson. 2017. Building on word animacy to determine coreference chain animacy in cultural narratives. In *The AIIDE-17 Workshop on Intelligent Narrative Technologies WS-17-20*.

Heng Ji and Dekang Lin. 2009. Gender and animacy knowledge discovery from web-scale n-grams for unsupervised person mention detection. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 1*, volume 1.

Folgert Karsdorp, M Meulen, Theo Meder, and APJ van den Bosch. 2015. Animacy detection in stories. In *Proceedings of the 6th Workshop on Computational Models of Narrative (CMN'15)*, pages 82–97. Atlanta, GA.

Seppo Kittila, Katja Vasti, and Jussi Ylikoski. 2011. *Introduction to Case, animacy and semantic roles*. John Benjamins Publishing Company.

Seppo Kittilä. 2006. Object-, animacy-and role-based strategies: A typology of object marking. *Studies in Language. International Journal sponsored by the Foundation Foundations of Language*, 30(1):1–32.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Heeyoung Lee, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2013. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4):885–916.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL): System Demonstrations*, pages 55–60. Baltimore, MD.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. http://arxiv.org/abs/1301.3781.

Joshua Moore, Christopher JC Burges, Erin Renshaw, and Wen-tau Yih. 2013. Animacy detection with voting models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 55–60.

Anders Nøklestad. 2009. *A Machine Learning Approach to Anaphora Resolution Including Named Entity Recognition, PP Attachment Disambiguation, and Animacy Detection*. Ph.D. thesis, University of Oslo, Oslo, Norway, May.

Constantin Orăsan and Richard Evans. 2001. Learning to identify animate references. In *Proceedings of the 2001 Workshop on Computational Natural Language Learning (CoNLL)*, page Article No. 16. Toulouse, France.

Constantin Orăsan and Richard J Evans. 2007. NP animacy identification for anaphora resolution. *Journal of Artificial Intelligence Research*, 29(1):79–103.

Lilja Ovrelid. 2005. Animacy classification based on morphosyntactic corpus frequencies: some experiments with Norwegian nouns. In *Proceedings of the Workshop on Exploring Syntactically Annotated Corpora*, pages 24–34. Birmingham, England.

Martha Palmer, Olga Babko-Malaya, and Hoa Trang Dang. 2004. Different sense granularities for different applications. In *Proceedings of the 2nd International Workshop on Scalable Natural Language Understanding (ScaNaLU 2004) at HLT-NAACL 2004*.

Martha Palmer, Hoa Trang Dang, and Christiane Fellbaum. 2007. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineering*, 13(2):137–163.

Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language*. Longman, London.

Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501. Association for Computational Linguistics.

Mark Saunders, Philip Lewis, and Adrian Thornhill. 2009. *Research methods for business students*. Prentice Hall,UK.

Sam Joshua Wiseman, Alexander Matthew Rush, Stuart Merrill Shieber, and Jason Weston. 2015. Learning anaphoricity and antecedent ranking features for coreference resolution. In *Association for Computational Linguistics and International Joint Conference on Natural Language Processing,*. Association for Computational Linguistics.

Mutsumi Yamamoto. 1999. *Animacy and reference: A cognitive approach to corpus linguistics*. John Benjamins Publishing, Amsterdam.