The AIIDE-17 Workshop on
Intelligent Narrative Technologies
WS-17-20

Building on Word Animacy to Determine Coreference Chain Animacy in Cultural Narratives

Labiba Jahan, Geeticka Chauhan, Mark A. Finlayson
School of Computing and Information Sciences
Florida International University
11200 S.W. 8th Street, ECS Building, Miami, FL 33199
{ljaha002, gchau005, markaf}@fiu.edu

Abstract

Animacy is the characteristic of being able to independently carry out actions in a story world (e.g., movement, communication). It is a necessary property of characters in stories, and so detecting animacy is an important step in automatic story understanding. Prior approaches to animacy detection have conceived of animacy as a word- or phrase-level property, without explicitly connecting it to characters. In this work we compute the animacy of referring expressions using a statistical approach incorporating features such as word embeddings on referring expression, noun, grammatical subject and semantic roles. We then compute the animacy of coreference chains via a majority vote of the animacy of the chain's constituent referring expressions. We also reimplement prior approaches to word-level animacy to compare performance. We demonstrate these results on a small set of folktales with gold-standard annotations for coreference structure and animacy (15 Russian folktales translated into English). Folktales present an interesting challenge because they often involve characters who are members of traditionally inanimate classes (e.g., stoves that walk, tree that talk). We achieve an F_1 measure 0.90 for the referring expression animacy model, and 0.86 for the coreference chain model. We discuss several ways in which we anticipate these results may be improved in future work.

Introduction

Characters are an indispensable element of narrative. Most definitions of narrative acknowledge the central role of character: Monika Fludernik, as just one example of many, defines a narrative as "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 . . . " (2009, p. 6). Thus, 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 such as props, locations, or other referents.

One first step toward character detection is *animacy* detection, where animacy is the characteristic of being able to independently carry out actions in a story world (e.g., movement or communication). 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 has conceived of animacy as a word-level phenomenon, marking animacy as an independent feature on each individual word (e.g., Orăsan and Evans 2007, Bowman and Chopra 2012, Karsdorp et al. 2015). However, characters and other entities are expressed in texts as coreference chains made up of referring expressions (Jurafsky and Martin 2007), and so we need some way of computing animacy on the chains directly. One way of doing this is to combine word-level animacy markings—say, using majority vote—into animacy for referring expressions and coreference chains. We take this method as a baseline approach. 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. This is the approach we pursue here, which we find has better performance.

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 to indicate animate entities, such as titles, animals, or personal pronouns (e.g., Quirk et al. 1985, Yamamoto 1999). However, stories 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 this work our task is to predict the animacy of referring expressions and coreference chains in stories. This is a preliminary study, and we only use a small corpus of 15 folktales to demonstrate the feasibility of the approach. We first annotated animacy on coreference chains directly, and then propagated these markings to the referring expressions. Using these annotations we then trained a support vector machine (SVM) classifier for the animacy of referring expressions themselves, and compared two methods for computing the animacy of a coreference chain using those values. Majority voting performed best in this context, and it outperforms a baseline that computes referring expression animacy by majority vote over the word-level animacy mark-

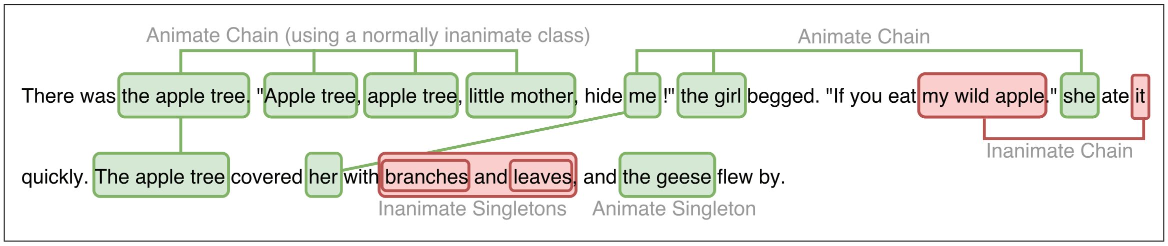

Figure 1: Example text containing animate and inanimate coreference chains. Colored boxes represent referring expressions, while links between them signify coreference. Animate chains are shaded green, while inanimate chains are shaded red. The text is drawn from Story #113 *The Magic Swan Geese* (Guterman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

ings. Overall we built three different models for animacy detection. The first is the referring expression model, on which the second model for coreference chains builds. We also built a third model for word-level animacy, which is used for our baseline comparison.

The paper is organized as follows. We first describe how we carried out the annotation of the data, and then describe the experimental setup, including features extracted, the training of the SVM, and how the different models were configured. We next present our results and discuss their significance. We finally outline related work which served as the inspirations and comparisons for this study, and conclude with a list of our contributions.

Data

Our data was a small corpus of Russian folktales that we assembled in the context of other work (Finlayson 2017). We started the project seeking to use existing data, as there have been a number of studies of animacy detection already (as we discuss in Related Work, below). However, no prior data in English was readily available to use. The best performing prior work was Karsdorp et al. (2015 which was done on a corpus of 74 stories comprising 74,504 words in Dutch. Orăsan and Evans (2007) did their work in English but their data was not readily available.

Our corpus contains 15 tales, originally collected in Russian in the late 1800's but translated into English in the mid-twentieth century. Table 1 summarizes counts of various aspects of the annotated data. The corpus contains gold-standard annotations for token and sentence boundaries, parts of speech (Penn Treebank II Tagset; Marcus, Marcinkiewicz, and Santorini 1993), referring expressions, and coreference chains (as well as other layers of annotation).

We annotated these tales for coreference- and word-level animacy. The annotation was done 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 animacy of referring expressions were directly calculated from the animacy of the coreference chains: if a coreference was marked as animate, all of its constituent referring expressions were also marked animate.

	Token	**Referring Expressions**	**Coref. Chains**
Total	23,291	6,631	1,633
Animate	3,896	4,288	344
Inanimate	19,395	2,343	1,289
Unique Items			
Animate	291	798	-
Inanimate	2221	1459	-
Total	2,199	2,231	-
Tokens	**Noun**	**Pronoun**	**Adjective**
Animate	1,658 (43%)	2,252 (58%)	38 (1%)
Inanimate	2,220 (11%)	401 (2%)	862 (4%)

Table 1: Counts of various aspects of annotated data, including total number of animate and inanimate tokens, referring expressions, and coreference chains, with breakdowns of number of unique items and part of speech in each class.

We also 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). This annotation was performed following two rules. First we marked as animate 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). We also 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 *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:

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

Table 2: 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.

> *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. (Tale #140, Guterman 1975, p. 458)*

A summary of examples for animate and inanimate words is given in Table 2.

Experimental Setup

Features

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

1. **Word Embeddings (WE)**: We computed 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 (2017), and configured the skip-gram model with a minimum word frequency of 3, layer width (dimensions) of 300, a seed of 42, 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.

2. **Word Embeddings on Referring Expressions (WER)**: We calculated 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.

3. **Composite Word Embedding (CWE)**: We computed a composite 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. This feature explicitely captures the tendency of nouns to refer to animate entities.

6. **Grammatical Subject (GS)**: Animate references tend to appear as the grammatical subjects of verbs (Ovrelid 2005). We used dependency parses generated by the Stanford dependency parser (Manning et al. 2014) to check if 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; 2011) to compute semantic roles for all the verbs in the stories. We then checked whether a given referring expression contained an ARG0 for a verb (an exact match was not required), and encoded this as a boolean feature.

Classification Models

We implemented our classification models using SVM (Chang and Lin 2011), with a Radial Basis Function Kernel. We varied the features used to train the different models as shown in Table 3. We trained each model using cross validation, and report macroaverages across the performance on test folds.

Model	Feature Set	Acc.	Inanimate κ	Inanimate Prec.	Inanimate Rec.	Inanimate F_1	Animate κ	Animate Prec.	Animate Rec.	Animate 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**
Referring Expressions	Baseline MFC	37%	0	0.38	1.0	0.55	0	0	0	0
	Baseline Maj. Vot.	75%	0.53	0.59	0.99	0.74	0.53	0.99	0.62	0.76
	WER	72%	0.49	0.58	0.99	0.73	0.49	0.98	0.57	0.72
	N	80%	0.56	0.85	0.60	0.70	0.56	0.80	0.93	0.86
	GS	80%	0.56	0.85	0.60	0.70	0.56	0.79	0.93	0.86
	SS	76%	0.51	0.67	0.74	0.70	0.51	0.83	0.78	0.80
	WER, GS	84%	0.64	0.89	0.66	0.76	0.63	0.82	0.95	0.88
	WER, SS	87%	0.72	0.87	0.79	0.82	0.70	0.87	0.91	0.89
	N, GS, SS	80%	0.56	0.84	0.60	0.70	0.56	0.79	0.93	0.86
	WER, N, GS	84%	0.64	0.88	0.67	0.76	0.64	0.82	0.95	0.88
	WER, N, GS, S	87%	0.73	0.85	0.80	0.83	0.71	0.88	0.90	0.89
	WER, N, SS	86%	0.70	0.83	0.77	0.80	**0.68**	**0.87**	**0.91**	**0.90**
Coreference	Maj. vote (all)	79%	0.48	0.93	0.80	0.86	0.48	0.50	0.76	0.61
	Maj. vote (long only)	84%	0.68	0.86	0.78	0.82	**0.68**	**0.82**	**0.89**	**0.86**

Table 3: 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.

We have three models for animacy: referring expressions, coreference chains, and words. For our referring expression animacy model, we explored different combinations of the 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 $p = 1$, which were chosen after a small amount of parameter space exploration. The first two values are relatively low in the range for these parameters, which is appropriate for a balanced class situation. We measured the performance of the classifier using 10-fold cross validation.

We calculated two baselines for referring expression animacy. The first is the majority class baseline (inanimate is the majority class). The second combines word-level animacy predictions generated by our word animacy model (discussed below) via a majority vote.

For the coreference chain animacy model, we implemented two majority vote approaches for combining the results of the referring expression animacy model to obtain a coreference animacy prediction. First, we computed the majority vote considering all referring expressions in a coreference chain. In the case of ties, the chain was marked inanimate. Because short coreference chains were responsible for much of the poor performance, we also calculated the performance of majority voting excluding chains of length four and below.

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 $p = 1$, which were chosen after a small amount of parameter space exploration. The first two values are relatively high in the range for these parameters, which is appropriate for a unbalanced class situation. 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. Our model achieved F_1 of 0.98 for the inanimate class, where the state of the art achieved 0.99. On the other hand, our model achieved an F_1 of 0.90 for the animate class, where the state of the art achieved 0.93.

Results & Discussion

We evaluated our models by measuring accuracy, precision, recall, F_1, and Cohen's kappa (κ) relevant to the gold-standard annotations. Table 3 summarizes the results for both the animate and inanimate classes. In the case of referring expression animacy we omit some combinations of features (e.g., WER & N) that produced especially poor results. We obtained the best result using three features: word embeddings over referring expressions (WER), noun (N) and semantic subject (SS). For the coreference animacy model, majority vote does not work as well as expected, with an overall F_1 of 0.61 when calculated over all chains. This poor performance relative to the word and referring expression animacy models is due largely to under-performance on short coreference chains (those with four referring expressions or fewer). This suggests that in future work we need to concentrate our effort on solving the short chain issue. We discuss this in more detail below.

Nevertheless, there is no prior work that reports animacy classification results directly for referring expressions and coreference chains, and so these results set the initial foundation for animacy classification of these objects.

Error Analysis & Future Work

A detailed error analysis of the results revealed at least four major problems for the classifier that we will focus on in future work: short chains, quotations, agency selection restrictions, and proper names.

Determining the animacy of short coreference chains is apparently a challenging task for our system. As the length of a chain tends toward a single referring expression, the coreference classifier performance 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 one approach to solving this problem is more data, and explicitly incorporating the animacy of heads of noun phrases as features.

The second problem is that 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 but the classifier detects it as animate because it finds three animate words "tsar", "daughter" and "dragon" in that quote. This will require some rule-based processing to address.

A third problem is that although animacy correlates with semantic subject position, it is not strictly implied by it. Consider the difference between "The bird flew across the field" (implies that *the bird* is animate) and "The ball flew across the field" (the ball is inanimate). To address this problem, we plan to incorporate animacy selectional restrictions as training features, where the selectional restrictions are drawn from existing lexical resources (e.g., VerbNet; Schuler 2005). This will allow us to distinguish between semantic roles which imply animacy and those which do not.

Finally, in the folktales we see names whose surface form are identical to inanimate entities, e.g., *Evening*, *Midnight*, or *Dawn*, as mentioned previously. Addressing this will requiring integrating named entity recognition into the system.

Related Work

Animacy Detection in English

Evans and Orăsan (2000) first explored animacy classification as a means to improve anaphora resolution. Their approach involves the identification of WordNet hypernym branches that should be always marked animate. They took this work forward by using a supervised machine learning (ML) method to mark unseen WordNet senses by their animacy (Orăsan and Evans 2001). They also explored both rule-based and machine-learning-based for animacy classification of nouns (Orăsan and Evans 2007). The rule-based method uses the unique beginners in WordNet for classification, while the machine learning method uses a multiple-step procedure to determine noun animacy. First, they use a statistical chi-squared method to determine the animacy of a sense (even for those not previously found in the annotated corpus, but which are hyponyms of a node which has been classified). For nouns whose sense was not known, they used machine learning for classification. They performed both intrinsic evaluation (achieving F_1 of 0.94 for animate class on one of the two corpus they use), but also extrinsic evaluation by measuring the impact of animacy detection on the performance of the MARS anaphora resolution system and a word sense disambiguation algorithm.

Bowman and Chopra (2012) conceived of animacy and inanimacy classification as a multi-class problem applied directly to noun phrases (NPs), using a maximum entropy classifier to classify NPs as *human*, *vehicle*, *time*, *animal*, etc, with an overall accuracy of 85%. Each class was considered ultimately animate or inanimate, meaning that a binary animacy classification could be derived from the marking. They achieved an overall accuracy of 94% for the binary animacy classification, but do not report F-measure statistics.

Animacy Detection in Other Languages

Nøklestad (2009) implemented animacy detection for Norwegian nouns, levering this along with Named Entity Recognition (NER) to improve the performance of anaphora resolution. They explored various ways to use data from the web to extract lists of animate nouns as well as to check the animacy value of a particular noun. For example, if the noun co-referred frequently with *han* (he) and *hun* (she), then it was characterized as animate. This is basically a rule-based method using queries to figure out the animacy of nouns. This method achieves an accuracy of 93%. The main problem with this approach, 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 (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* 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 lemma), syntactic features (dependency parse to check which word is nsubj or nobj), morphological features (POS tags), and semantic features (word embedding using 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 + POS + embedding features.

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 directly, which is a necessary step for reliably detecting characters in stories.

Contributions

We built a system for animacy classification of referring expressions and coreference chains to move towards automatic character detection within stories. Our work provides several contributions. First, we annotated 15 Russian folktales translated into English for animacy information at the word-level and coreference-chain level. Second, we implemented an SVM classifier using features inspired by previous work to predict the animacy of referring expressions directly, achieving good performance of 0.90 F_1. Finally, we used a majority voting approach to obtain the animacy of coreference chains. The overall performance of this approach was poorer than expected, at 0.61 F_1, but error analysis suggested several potential ways forward to improving that performance, in particular, focusing on the animacy classification of short chains (i.e., chains with four or fewer referring expressions). Measuring the performance of the majority voting approach on long chains (five or more referring expressions), revealed a much better performance of 0.86 F_1.

Acknowledgments

This work was partially supported by National Institutes of Health (NIH) grant number 5R01GM105033-02. We would also like to thank Joshua Eisenberg and W. Victor H. Yarlott from the FIU Cognac Lab for assisting us with issues related to setting up software.

References

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

Bowman, S. R., and Chopra, H. 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*, 7–10. Montreal, Canada.

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

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.

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

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

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

Finlayson, M. A. 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.

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

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

Jurafsky, D., and Martin, J. H. 2007. *Speech and Language Processing*. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Karsdorp, F.; Meulen, M.; Meder, T.; and van den Bosch, A. 2015. Animacy detection in stories. In *Proceedings of the 6th Workshop on Computational Models of Narrative (CMN'15)*, 82–97. Atlanta, GA.

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

Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S. J.; and McClosky, D. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL): System Demonstrations*, 55–60. Baltimore, MD.

Marcus, M. P.; Marcinkiewicz, M. A.; and Santorini, B. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics* 19(2):313–330.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. http://arxiv.org/abs/1301.3781.

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

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

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

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

Quirk, R.; Greenbaum, S.; Leech, G.; and Svartvik, J. 1985. *A Comprehensive Grammar of the English Language*. London: Longman.

Schuler, K. K. 2005. *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. Dissertation, University of Pennsylvania.

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