Florida International University
School of Computing and Information Sciences

Research and Software Engineering Focus

Final Deliverable

Character Identification and Classification

Team Members: Geeticka Chauhan, Labiba Jahan (volunteer)

Product Owner(s): Mark Finlayson

Mentor(s): Francisco Ortega, Mohsen Taheri

Instructor: Masoud Sadjadi

Abstract

This document presents the information necessary to gain a good understanding of the research project involving Character Identification and Classification. Our aim is to perform automatic classification of characters in folktales by using the concept of animacy. We define animacy as the property of being alive and communicative, for example: people and animals. In addition to these entities, folktales often involve characters that are not seen as animate in the real world, and this makes our problem a challenging one. Unlike other work, we attempt to classify the animacy of coreference chains by the use of a multi-sieve approach by combining a rule-based method and classifier. Our results are promising so far, and reaching very close in F1 measure of the state of the art techniques.

Table of Contents

INTRODUCTION

This section provides a brief introduction to our research problem and motivations. First, the goals of the project have been described, followed by terminology and the process followed in the execution of this project.

Goals

Our research project's goal is to understand stories and to create a science behind story understanding. The tool that we use for our project is Natural Language Processing (NLP). Examples of NLP include the fake news detector chrome plug-in known as Slate (Fig. 1) and IBM Watson for Genomics (Fig. 2) Stories can be divided into plot and characters (Fig. 3), and in order to extract characters, we leverage the concept of animacy. We define animacy as the quality of being alive and communicative, for example, people and animals as well as normally marked inanimate characters such as talking stoves in fairytales. We take animate entities as personal and non-personal noun phrases. Personal noun phrases are classified by male, female, dual, common and collective, whereas non-personal noun phrases are classified by common, collective, higher or lower nouns. In short, we include titles such as daughter, doctor, baby, family (collective), animals (but not insects) as well as talking inanimate entities such as talking trees (due to the fairytale nature of folktales). Our problem is a challenging one which cannot be solved by simple Named Entity Recognition (NER) systems because of the unconventional characters existing in our corpus of stories.

Fig. 1: Slate

Fig. 2: IBM Watson

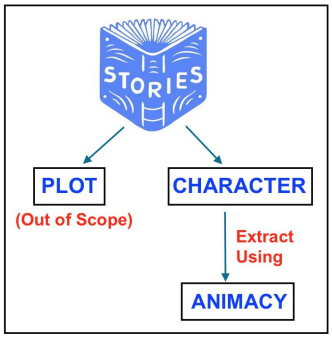

Fig. 3: Stories

Terminology

We measure animacy for coreference chains within our corpus of stories. Coreference chains are chains of referring expressions.

Referring Expression: Multiple sets of words that refer to the same entity in the real world. In Fig. 4, the text highlighted in yellow is referring expressions.

Coreference Chains: Chain of referring expressions shown in Fig. 4 as the chain of green boxes.

Animate Chains: Marked in Fig. 4 for chains.

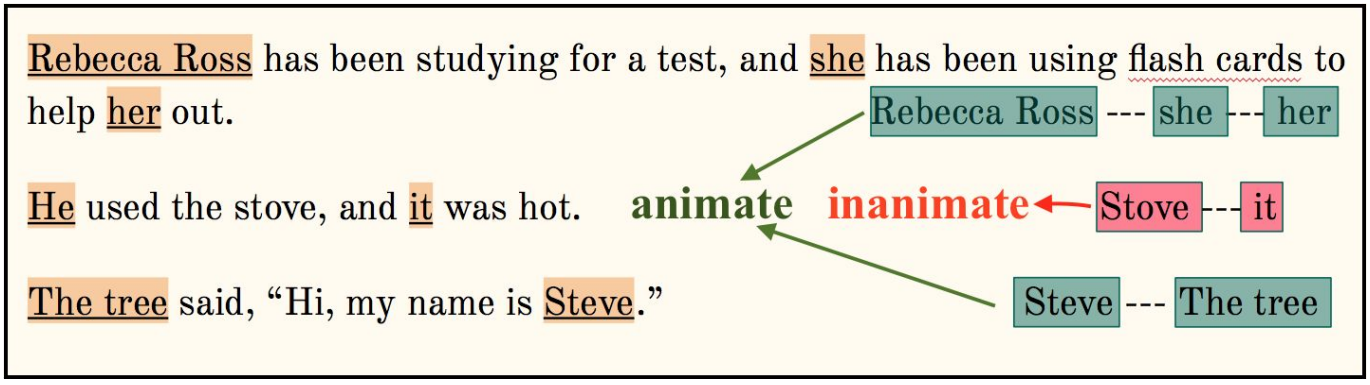

Fig. 4: Animacy of Coreference Chains

Process

The process that we follow throughout our research (in Fig. 5) is that of a rule-based as well as a machine learning based system combined. We have implemented the machine-learning aspect using a Support Vector Machine (SVM), and we are still in the process of deciding the rules for our rule-based system. We use the input as 15 annotated Russian Folktales with the NLP pipeline and hand-annotate their coreference chains according to animacy. We divide these into 10 for training and 5 for testing and get results using various combinations of features. Our whole experimental setup is explained in Appendix D with the draft research paper. The results we get are in Fig. 6 and after doing various performance tests, we came to the conclusion that the model on referring expressions with the word and SRL head features work the best (refer to Appendix D for explanation).

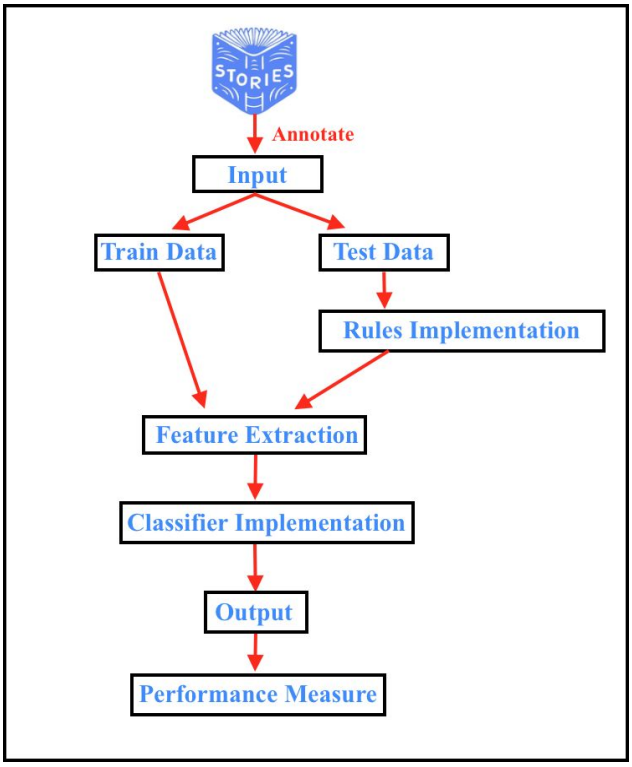

Fig. 5: Process of Research

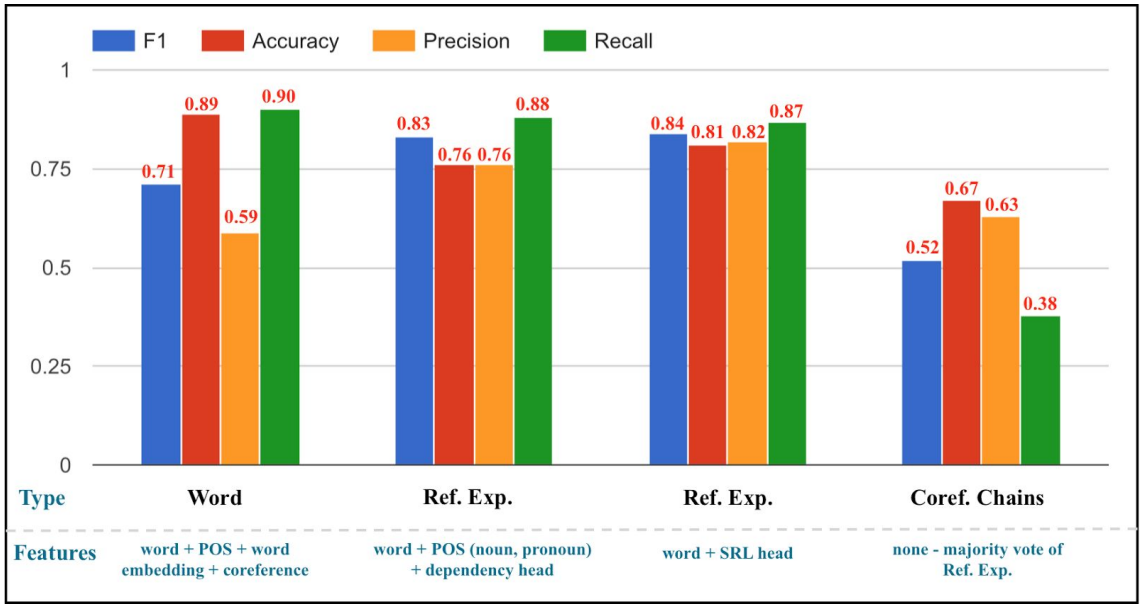

Fig. 6: Results

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the Character Identification and Classification project. These user stories served as the basis for the implementation of the project's research. This section also shows the user stories that are to be considered for future research.

List of User Stories

User Story #	User Story Name	Information
107	Reading Animacy Papers + Books	Read NP Animacy Identification for Anaphora Resolution, Improving Anaphora Resolution by identifying animate entities in text, Animacy Detection in Stories, Morphology of the Folktale
125	Set up Stanford NLP + Extract Subjects from Dependency Parse	Set up stanford NLP and extracted subjects by sentence (with token number) and without token number to feed as a feature to the SVM
130	Set up SRL and Extract ARG0	Set up SRL and got the ARG0 ie the semantic subject (as opposed to grammatical subject) of the text to feed as a feature to the SVM.
126	Extract Modifier Properties from WordNet	Extract amod adjective modifier from dependency parse and calculate word distance with known animate senses to get an idea of the animacy of the noun phrase it modifies; also get the animacy of a noun phrase using wordnet
151	Annotate more Stories + Error Analysis on Referring Expressions	Annotate story # 16-26, extract the false positives and false negatives for referring expressions in 1-15 and do error analysis
159	Create Rules for the Model	Extract the heads of each referring expression for stories 1-15.
147	Calculate Fa and Fm + Implement SVM using Fa and Fm as features	Extract features related to the positioning of referring expressions in order to implement SVM on coreference chain
166	Review annotation of Stories	Improve the annotation to improve results
167	Finalize Rules + Extract POS of head	Finish rules and extract part of speech

Implemented User Stories

User Story # 107

Description:

- As a researcher, I would like to be able to understand the background on animacy so that I am able to make sure that the research is unique.

Process:

1. Read the paper "NP Animacy Identification for Anaphora Resolution" – helped me understand Rule-Based and well as Machine Learning Methods to perform animacy classification and see how it can be applied to improve anaphora resolution.
2. Read the paper "Improving anaphora resolution by identifying animate entities in text", "Learning to identify animate references" and "A machine learning approach to anaphora resolution including named entity recognition, PP Attachment Disambiguation, and Animacy Detection" to help confirm that most research is being done to use animacy to improve anaphora resolution, which is different from my research in the sense that only word animacy is looked at.
3. Read "Animacy Detection in Stories" to understand the process of how to calculate word animacy using a machine learning method.
4. Read the book "Morphology of the Folktale" to get information about linguistic aspects behind the research.

User Story # 125

Description:

- As a researcher, I want to set up Stanford Core NLP and extract the dependency parse of each story so that I can provide the output with the subjects to Labiba so that she can use them as features to use in the SVM for finding the animacy of referring expressions

Acceptance Criteria:

1. Must use 15 text files of the Stories as input.
2. Must generate an xml file for the Dependency Parse of each Story.
3. Must extract the dependent inside the type "nsubj" dependency and place the grammatical subjects into a txt file.

Use Case:

- Name: Extract Subjects from Dependency Parse
- Actor: Researcher
- Preconditions: Must have the 15 txt files of stories as input to the dependency parse
- Flow of Events:

Actor Steps	System Steps
1. The researcher must call the DependencyParse.GenerateParse() method to run the Stanford NLP pipeline.	2. The System outputs the Stanford NLP pipeline for each story.
3. The researcher must call the ExtractSubject.Extract() method.	4. The System outputs the subjects by sentence and annotated with their token number.
5. The researcher must call the ExtractSubjectByWord.Extract() method.	6. The System outputs the subjects in the overall file, removing duplicates and not specifying the token number.

- Exit Condition: This use case ends when subjects have been extracted according to their token and sentence number as well as by removing duplicates.

User Story # 130

Description

- As a researcher, I want to extract the ARG0 from the SRL parse so that I can add it as a feature in the SVM.

Acceptance Criteria

1. Must use the XML files of Stories to extract separate sentences.
2. Must have sentences in a separate txt file for each story.
3. Must extract the ARG0 from every sentence (if present) in the txt file for each story

Use Case:

- Name: Extract Semantic Subjects through SRL
- Actor: Researcher
- Preconditions: Must have the 15 XML files of stories as input to the sentence extractor
- Flow of Events:

Actor Steps	System Steps
1. The researcher must call the ExtractSentence.Extract() method.	2. The System outputs the first and last token number of each sentence.
3. The researcher must call the ExtractSentenceWords.Extract() method.	4. The System outputs each sentence per line for each of the stories.
5. The researcher must call tester.SRLrun() method.	6. The System outputs the semantic subjects, without removing duplicates and not specifying the token number.

- Exit Condition: This use case ends when semantic subjects have been extracted.

User Story # 151

Description:

- As a Researcher I would like to annotate more stories and do error analysis on the result of SVM so that I can decide how to improve our result for animacy of referring expressions

Acceptance Criteria:

- Must have the input as stories 16-26
- The stories should be xml files in the format created by Story Workbench
- Input for error analysis must be the list of false positive and false negative referring expressions for stories 1-15

Process:

- Annotated the stories 16-26 based on animacy of entire coreference chains (from which animacy of referring expressions is easy to extract)
- Wrote code inside the SVM code to get the referring expressions that were being characterized as false positives and those that were being characterized as false negative

User Story # 126

Description

- As a researcher, I want to extract modifier distances and utilize WordNet so that I can get more information about modifier animacy and the noun phrase's animacy respectively.

Acceptance Criteria

1. Must use the 15 XML files of dependency parse as input.
2. Calculate Modifier distances to get idea about animacy.
3. Write method to be able to tell whether a given noun is a living being.

Use Case

- Name: Calculate modifier distances and figure out whether given noun is a living being
- Actor: Researcher
- Preconditions: Must have the 15 XML files of stories as input to the sentence extractor
- Flow of Events:

Actor Steps	System Steps
1. The researcher must call the ExtractModifier.Extract() method.	**2.** The System outputs the modifiers along with the nouns they modify for every sentence. **4.** The System outputs the distance between the

3. The researcher must call the WordnetWork.Output().getModifierDistance() method. **5.** The researcher must call WordnetWord.Output().getAnimacy() method.	passed modifier and "aliveness" indicator of the modifier. **6.** The System outputs whether the passed noun phrase is animate.

- Exit Condition: This use case ends when information has been gotten about the animacy of noun phrase, modifier information and the list of modifiers from the stories.

User Story # 159

Description

- As a researcher, I want to work on the rule-based model so that we can have a supplemental model for the classifier.

Acceptance Criteria

1. Parse Referring Expressions with their sentence numbers
2. Get heads of Referring Expressions from Dependency Parse

Use Case

- Name: Extract heads of each referring expressions
- Actor: Researcher
- Preconditions: Must have the 15 XML files of stories as input
- Flow of Events:

Actor Steps	System Steps
1. The researcher must call the RefHeads.GenerateRefExp1() method. 3. The researcher must call the RefHeads.Run() method.	2. The System outputs the referring expressions for each story along with their sentence numbers according to story workbench and stanford NLP. 4. The System outputs an xml file for each story with the referring expression, its ID, its sentence number according to story workbench and stanford NLP along with the head of the referring expression.

- Exit Condition: This use case ends when the heads of referring expressions have been extracted.

Pending User Stories

User Story # 126

Description:

- As a researcher, I would like to finalize the rules and extract the part of speech of heads of referring expressions so that I can get better results.

User Story # 166

Description:

- As a researcher, I would like to review the annotation of the stories to make sure that the results we are getting are the best they can be.

User Story # 167

Description:

- As a researcher, I would like to finalize the rules and extract the part of speech of heads of referring expressions so that I can get better results.

PROJECT PLAN

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

1. Laptop Computer with at least 8GB RAM
2. Eclipse IDE
3. Annotated Stories outputted by MIT Story Workbench
4. Word2Vec Library in Java
5. MIT Java WordNet Interface
6. Semantic Role Labeler developed by Dr. Mark Finlayson
7. Stanford CoreNLP

Sprints Plan

Sprint 1

Date: 1/16
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00 pm
End time: 2:30 pm

After discussion, the velocity of the team were estimated to be 24 points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #107 Reading Animacy Papers + Books

The team members indicated their willingness to work on the following user stories.

- Geeticka Chauhan
    - User Story #107 Reading Animacy Papers + Books

Sprint 2

Date: 1/30
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00 pm
End time: 2:30 pm

After discussion, the velocity of the team were estimated to be 24 points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #125 Set up Stanford NLP + Extract Subjects from Dependency Parse

The team members indicated their willingness to work on the following user stories.

- Geeticka Chauhan
    - User Story #125 Set up Stanford NLP + Extract Subjects from Dependency Parse

Sprint 3

Date: 2/13
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00 pm
End time: 2:30 pm

After discussion, the velocity of the team were estimated to be 24 points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #130 Set up SRL and Extract ARG0

The team members indicated their willingness to work on the following user stories.

- Geeticka Chauhan
    - User Story #130 Set up SRL and Extract ARG0

Sprint 4

Date: 2/27
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00 pm
End time: 2:30 pm

After discussion, the velocity of the team were estimated to be 24 points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story # 151 Annotate more Stories + Error Analysis on Referring Expressions

The team members indicated their willingness to work on the following user stories.

- Geeticka Chauhan
    - User Story # 151 Annotate more Stories + Error Analysis on Referring Expressions

Sprint 5

Date: 3/13
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00 pm
End time: 2:30 pm

After discussion, the velocity of the team were estimated to be 30 points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #126 Extract Modifier Animacy Properties from WordNet

The team members indicated their willingness to work on the following user stories.

- Geeticka Chauhan
    - User Story #126 Extract Modifier Animacy Properties from WordNet

Sprint 6

Date: 3/27
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00 pm
End time: 2:30 pm

After discussion, the velocity of the team were estimated to be 30 points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #159 Create Rules for the Model

The team members indicated their willingness to work on the following user stories.

- Geeticka Chauhan
    - User Story #159 Create Rules for the Model

System Design

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

Architectural Patterns

The architectural pattern we follow is that of a 1-layer Object Oriented Architecture. We utilize the concepts of encapsulation and data hiding in order to separate the methods and variables used by different classes. Additionally, all our tasks have been separated as classes which create output that is independent of any other class. We chose this architecture because it was supported by our choice of language, Java. We chose Java due the availability of extensive NLP libraries in it such as Stanford coreNLP, MIT Java WordNet Interface etc., which provide very good results for the research problems they try to solve.

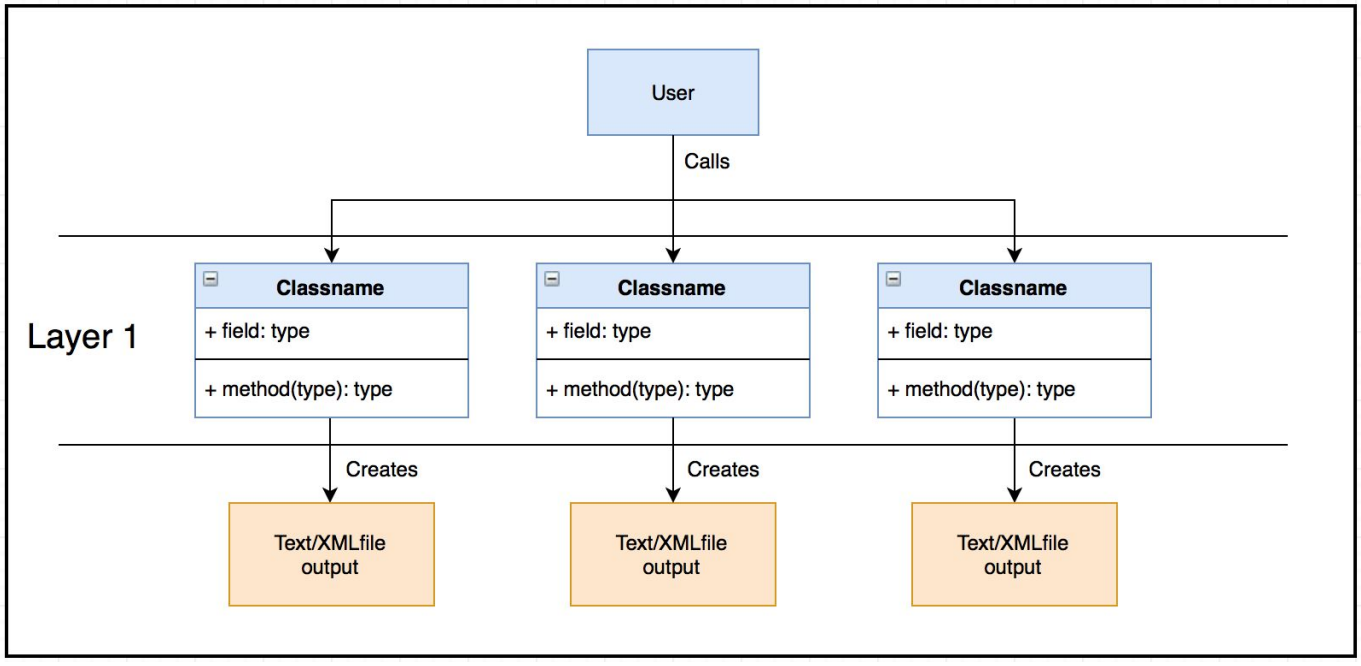

Fig. 7: Architectural Pattern of Object Oriented Design

System and Subsystem Decomposition

Our architectural pattern is described in Fig. 7, which shows our 1-layer Object Oriented Architecture and our system as a whole. Our subsystem is described by Fig. 8 with all the class names and components within the system. SRL and StanfordNLP are separate Java projects for feature extraction purposes whose outputs are fed into the SVM, which is a separate Java project implementing the SVM.

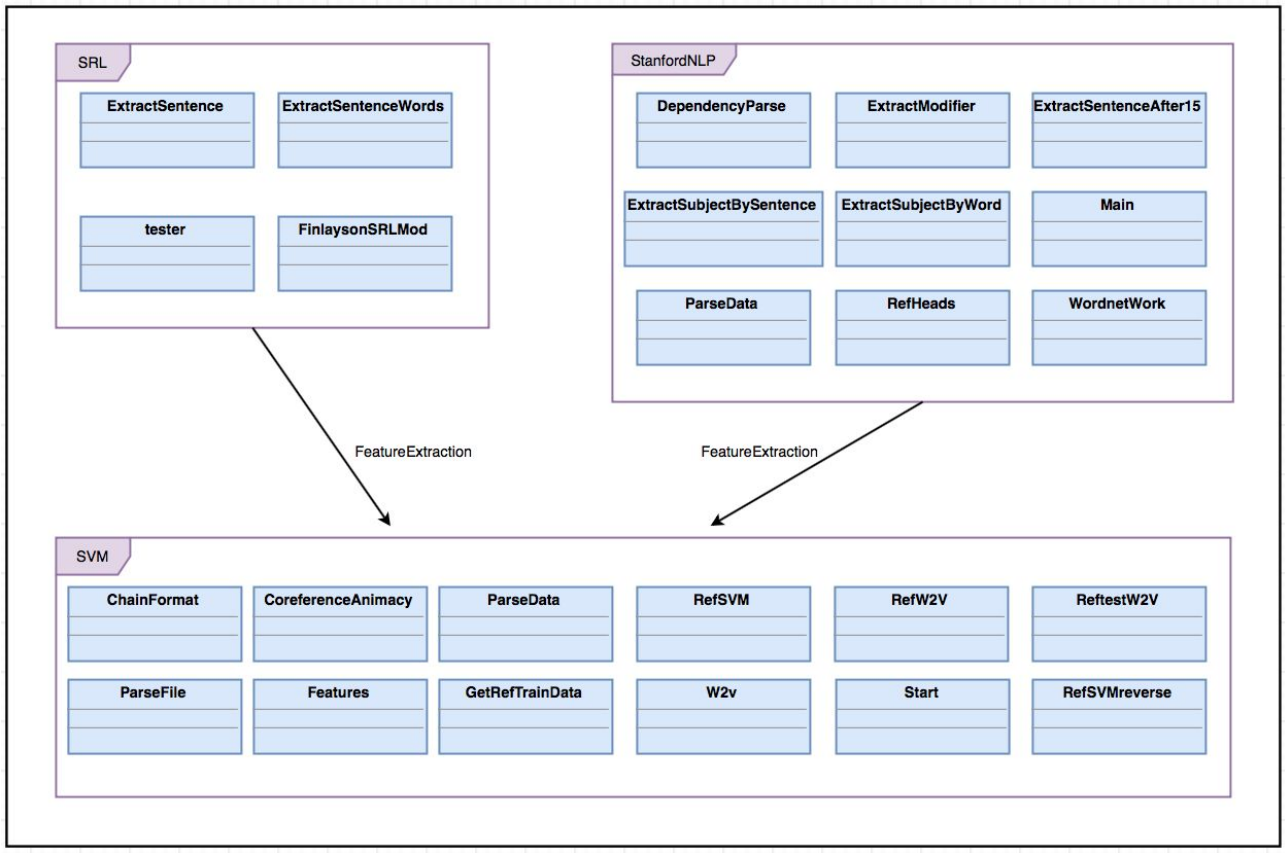

Fig. 8: Package Diagram of System and Subsystems

Deployment Diagram

The user goes into their Java Integrated Development Environment (IDE), imports the 3 Java projects and runs them in order to get the required output. These projects interact with the 15 Russian Folktales in our corpora in order to produce their outputs.

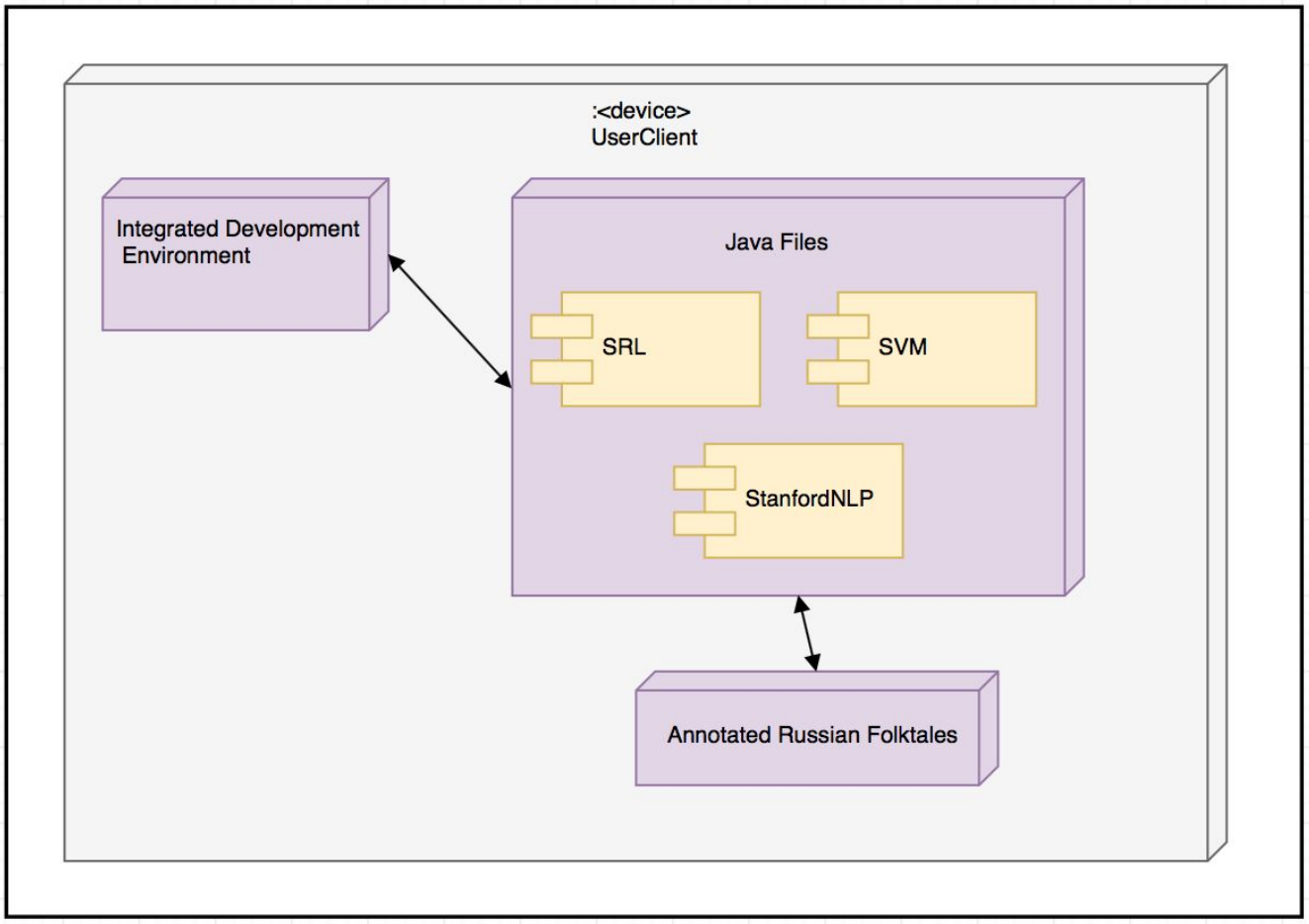

Fig. 9: Deployment Diagram

Design Patterns

The design patterns we use are the command and singleton design patterns.

We use command because we utilize the object-oriented architecture where every request is defined by objects. We divide our tasks as different classes which consist of methods that do related actions. For example, we have defined a class called “ParseData” that takes care of all the parsing of referring expressions and coreference chains from the input files. We designed many classes in such a way that they acted as black boxes that produced the desired output when one of their methods was ran.

We use singleton because most of the methods in our classes only need to be run once and do not need multiple objects being created. For example, we use the “DependencyParse.Extract()” method once and then move on to performing other tasks such as “ExtractSentence” which utilizes the output produced by the previous method.

SYSTEM VALIDATION

This section discusses the unit and integration tests we performed for different parts of our system. Every user story does not involve testing because this was a research project for which we needed to take some time to read papers and create short literature reviews for our final paper in Appendix D.

User Story # 125

Test Suite 125

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the GetParse function of DependencyParse class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 text files of the stories without annotation

Input:

- 15 text files of the stories without annotation

Expected Results:

- Complete NLP Pipeline for each of the 15 stories

Actual Result:

- Complete NLP Pipeline for each of the 15 stories

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the GetParse function of DependencyParse class

Pre-condition:

- The researcher should have access to the code
- The input must be text files of the stories

Input:

- 14 XML files with annotation

Expected Results:

- Complete NLP Pipeline for each of the 14 stories

Actual Result:

- Error of NoFileFound as the 15th story does not exist and is mentioned in the for loop

- Even when the for loop says 14, NLP Pipeline for text portion of file as well as the XML tags, which involves having to do a lot of cleaning up of the output.

Status (Fail/Pass):

- Fail

Integration Tests

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the Extract function of the ExtractSubjectByWord class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories with the NLP pipeline

Input:

- 15 XML files of the stories with the NLP pipeline

Expected Results:

- 15 text files with each unique subject on one line

Actual Result:

- 15 text files with each unique subject on one line

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the Extract function of the ExtractSubjectByWord class

Pre-condition:

- The researcher should have access to the code
- The input must be xml files of the stories

Input:

- 14 XML files without dependency parse in the NLP pipeline

Expected Results:

- Complete NLP Pipeline for each of the 14 stories

Actual Result:

- Error of NoFileFound as the 15th story does not exist and is mentioned in the for loop
- Even when the for loop says 14, there is no output generated (ie output files are empty) as no dependency parse is present

Status (Fail/Pass):

- Fail

User Story # 130

Test Suite 130

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the Extract function of ExtractSentence class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories without annotation

Input:

- 15 XML files of the stories with annotation

Expected Results:

- Start and end token of each sentence for each story

Actual Result:

- Start and end token of each sentence for each story

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the Extract function of ExtractSentence class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories without annotation

Input:

- 14 text files without annotation

Expected Results:

- Start and end token of each sentence for each story

Actual Result:

- Error of NoFileFound as the 15th story does not exist and is mentioned in the for loop
- Even when the for loop says 14, the tag dep wouldn't be found in the text files to be able to extract the starting and ending token.

Status (Fail/Pass):

- Fail

Integration Tests

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the SRLrun function of the tester class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories with the MIT Story Workbench pipeline

Input:

- 15 XML files of the stories with the MIT Story Workbench pipeline

Expected Results:

- 15 text files with subject on one line

Actual Result:

- 15 text files with each unique subject on one line

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the SRLrun function of the tester class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 text files of each sentence in the stories

Input:

- 14 XML files with annotation

Expected Results:

- 15 text files with each unique subject on one line

Actual Result:

- Error of NoFileFound as the 15th story does not exist and is mentioned in the for loop
- Even when the for loop says 14, no output is produced because only one sentence can be inputted at a time to the SRL

Status (Fail/Pass):

- Fail

User Story # 126

Test Suite 126

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the Extract function of ExtractModifier class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories in the stanford NLP format

Input:

- 15 XML files of the stories with annotation

Expected Results:

- Modifier and noun that it modifies for each story

Actual Result:

- Modifier and noun that it modifies for each story

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the Extract function of ExtractModifier class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories in Stanford NLP format

Input:

- 14 text files without annotation

Expected Results:

- Modifier and noun that it modifies for each story

Actual Result:

- Error of NoFileFound as the 15th story does not exist and is mentioned in the for loop
- Even when the for loop says 14, the tag dep wouldn't be found in the text files to be able to extract the starting and ending token.

Status (Fail/Pass):

- Fail

Integration Tests

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the Output function of the WordnetWork class

Pre-condition:

- The researcher should have access to the code

Input:

- Modifier: alive, noun phrase: table

Expected Results:

- Distance between modifier alive and “aliveness”
- Result of animacy of noun phrase table

Actual Result:

- Distance between modifier alive and “aliveness” ie 0.066
- Result of animacy of noun phrase “not alive”

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the Output function of the WordnetWork class

Pre-condition:

- The researcher should have access to the code

Input:

- Modifier: vital, noun phrase: dungon

Expected Results:

- Distance between modifier and “aliveness” as being more than 0
- Result of animacy of noun phrase

Actual Result:

- Distance of 0 between vital and aliveness
- Result of “word not existing” in wordnet for dungon due to error of spelling from dungeon.

Status (Fail/Pass):

- Fail

User Story # 159

Test Suite 159

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the GenerateRefExp1() function of RefHeads class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories in the story workbench format

Input:

- 15 XML files of the stories with annotation

Expected Results:

- Output format: RefExp & SentenceID ~ RefExpID&StanfordSentenceID where SentenceID is the ID with respect to Story Workbench and StanfordSentenceID is the ID with respect to Stanford Core NLP

Actual Result:

- RefExp & SentenceID ~ RefExpID&StanfordSentenceID

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the GenerateRefExp1() function of RefHeads class

Pre-condition:

- The researcher should have access to the code
- The input must be 15 XML files of the stories in the story workbench format

Input:

- 14 text without annotation

Expected Results:

- Output format: RefExp & SentenceID ~ RefExpID&StanfordSentenceID where SentenceID is the ID with respect to Story Workbench and StanfordSentenceID is the ID with respect to Stanford Core NLP

Actual Result:

- Error of NoFileFound as the 15th story does not exist and is mentioned in the for loop
- Even when the for loop says 14, the tag rep wouldn't be found in the text files to be able to extract the referring expressions.

Status (Fail/Pass):

- Fail

Integration Tests

Test case 1 (Sunny Day):

Description/Summary of Test:

- To test the output of the Run function of the RefHeads class

Pre-condition:

- The researcher should have access to the code
- The input must be output produced by GenerateRefExp1 function of the 15 files

Input:

- 15 text files generated by GenerateRefExp1 function

Expected Results:

- 15 xml files giving the heads of each referring expression

Actual Result:

- 15 xml files giving the heads of each referring expression

Status (Fail/Pass):

- Pass

Test case 2 (Rainy Day):

Description/Summary of Test:

- To test the output of the Run function of the RefHeads class

Pre-condition:

- The researcher should have access to the code
- The input must be output produced by GenerateRefExp1 function of the 15 files

Input:

- 14 xml files of story workbench format

Expected Results:

- 15 xml files giving the heads of each referring expression

Actual Result:

- Error of NoFileFound as the 15th story does not exist and is mentioned in the for loop
- Even when the for loop says 14, no output is produced because the story workbench format does not have the expected format of &, ~ etc.

Status (Fail/Pass):

- Fail

GLOSSARY

Natural Language Processing (NLP): Allowing computers to process human language through a variety of techniques such as Machine Learning (ML).

Referring Expression: Multiple sets of words that refer to the same entity in the real world. In Fig. 4, the text highlighted in yellow is referring expressions.

Coreference Chains: Chain of referring expressions shown in Fig. 4 as the chain of green boxes.

Animate Chains: Marked in Fig. 4 for chains.

Support Vector Machine (SVM): A machine learning classifier we use with the skip-gram model for word2vec.

Word2Vec: Conversion of words to vectors (a series of numbers) that keep the semantic information of the word intact.

WordNet: An ontology of words and their synonym, antonym etc relationships annotated by Princeton.

APPENDIX

Appendix A - UML Diagrams

User Story #107

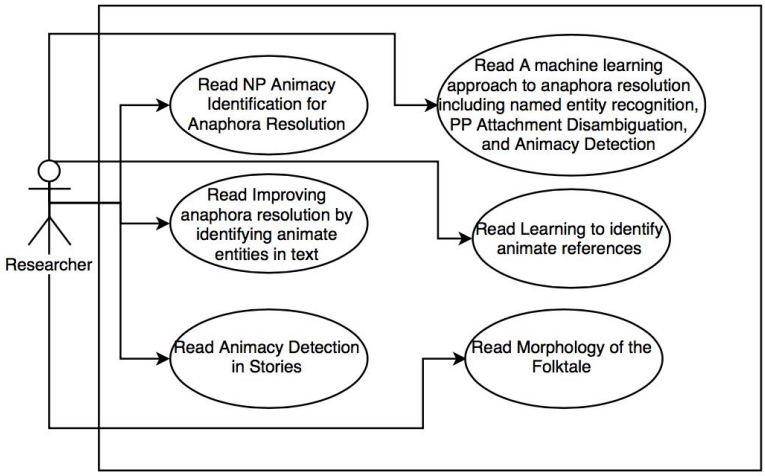

Fig. 10: Use Case Diagram # 107

User Story #125

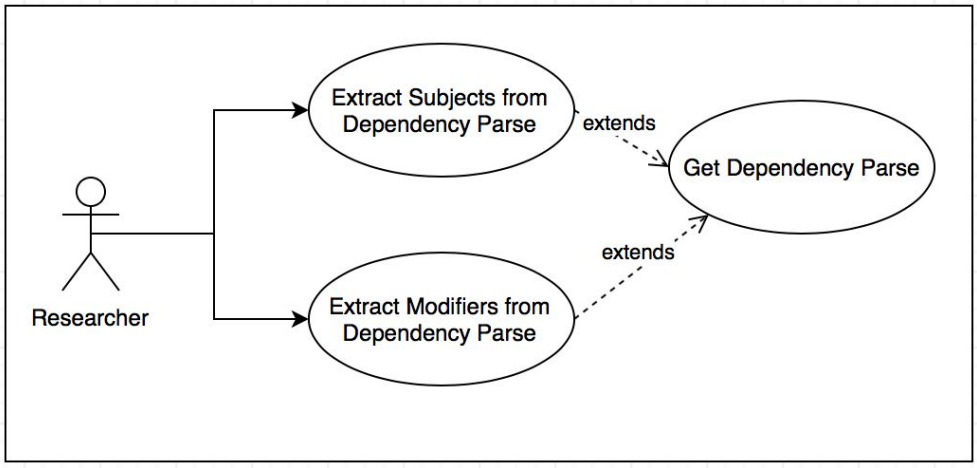

Fig. 11: Use Case Diagram # 125

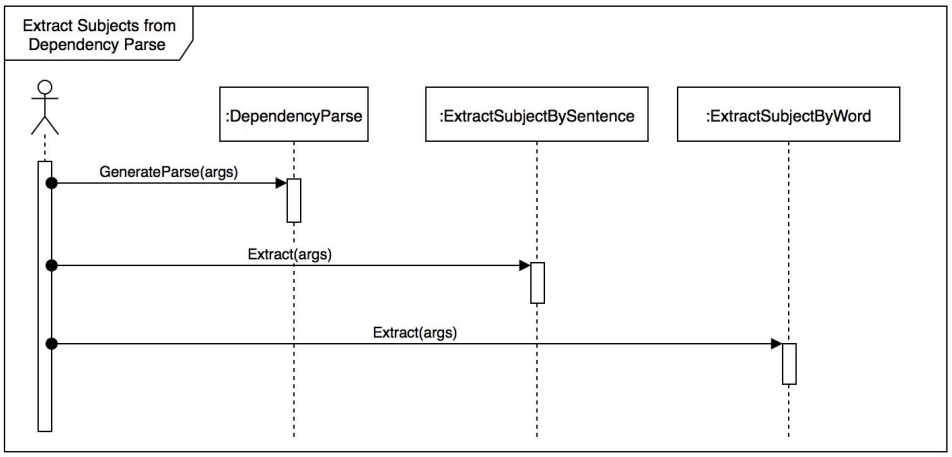

Fig. 12: Sequence Diagram # 125

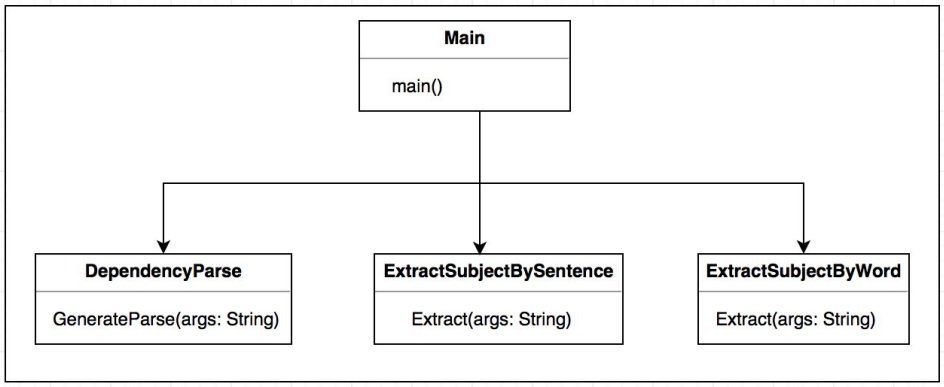

Fig. 13: Class Diagram # 125

User Story #130

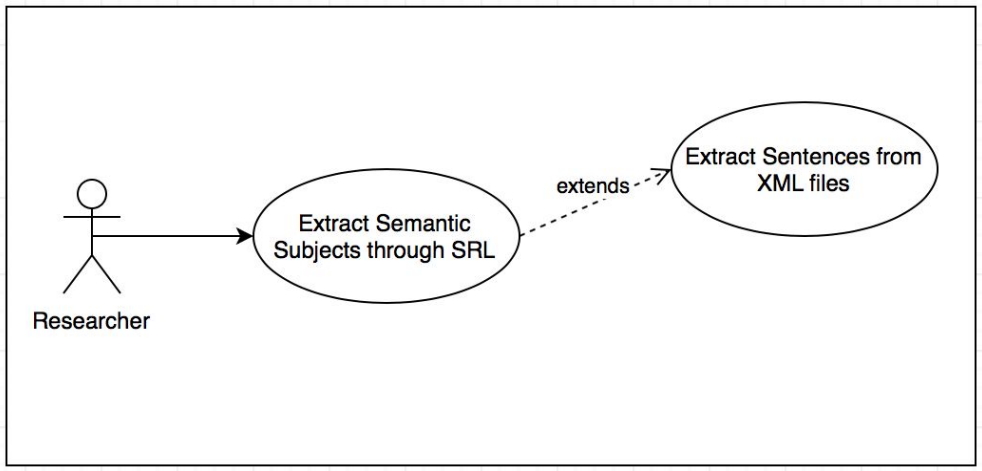

Fig. 14: Use Case Diagram # 130

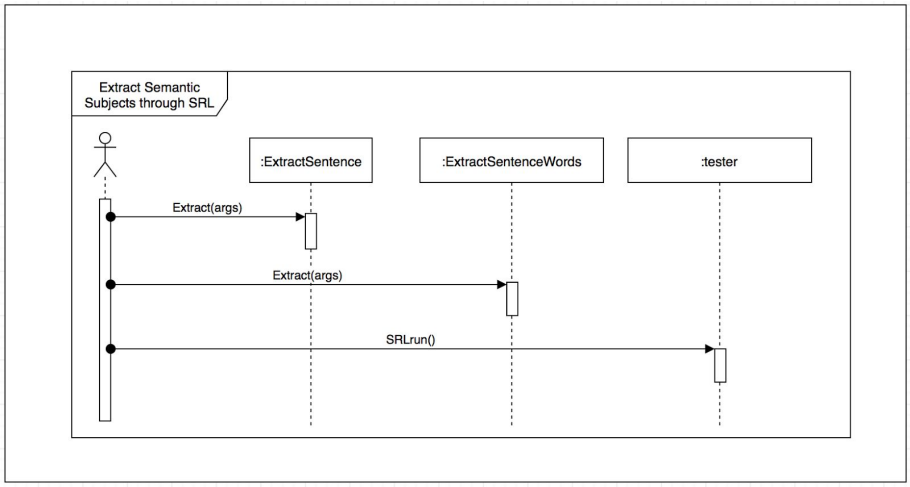

Fig. 15: Sequence Diagram # 130

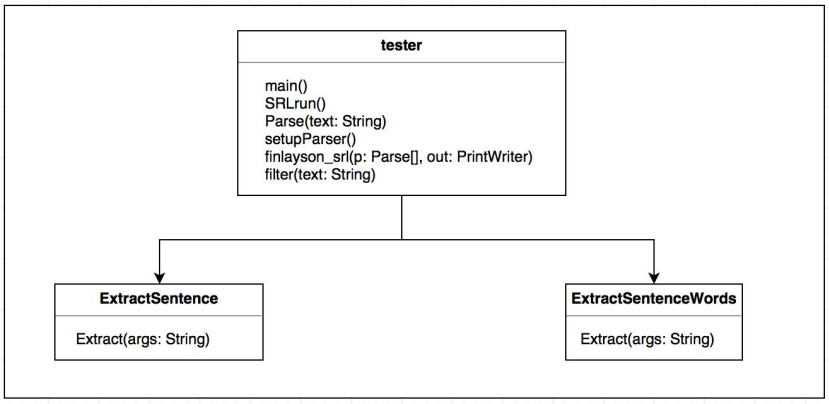

Fig. 16: Class Diagram # 130

User Story # 151

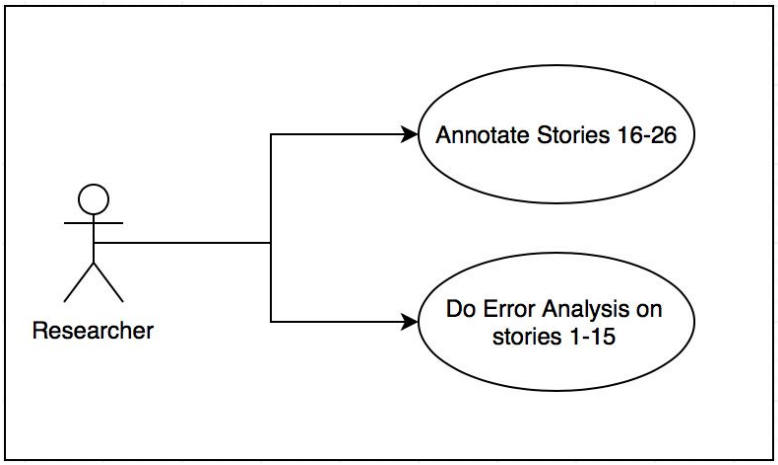

Fig. 17: Use Case # 151

User Story #126

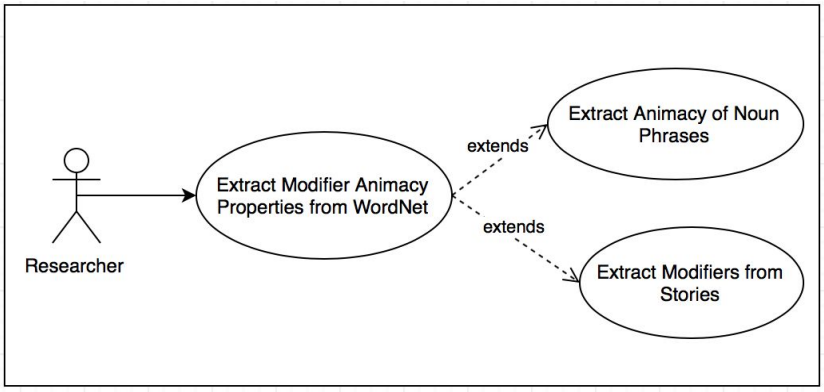

Fig. 18: Use Case # 126

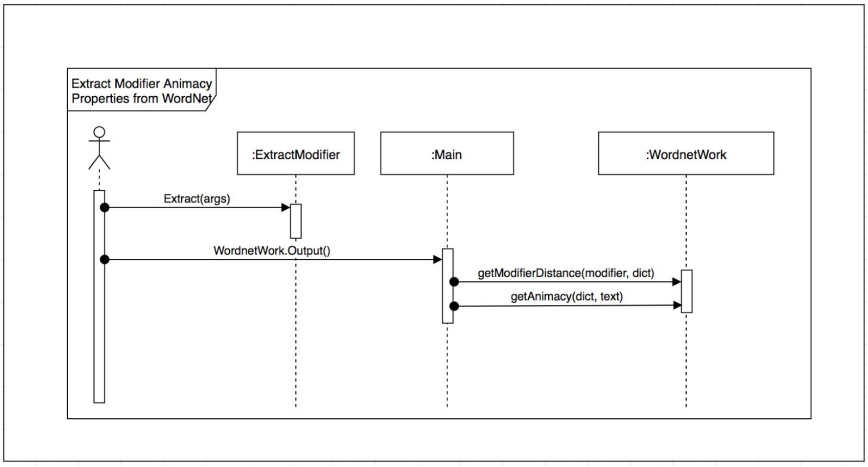

Fig. 19: Sequence Diagram # 126

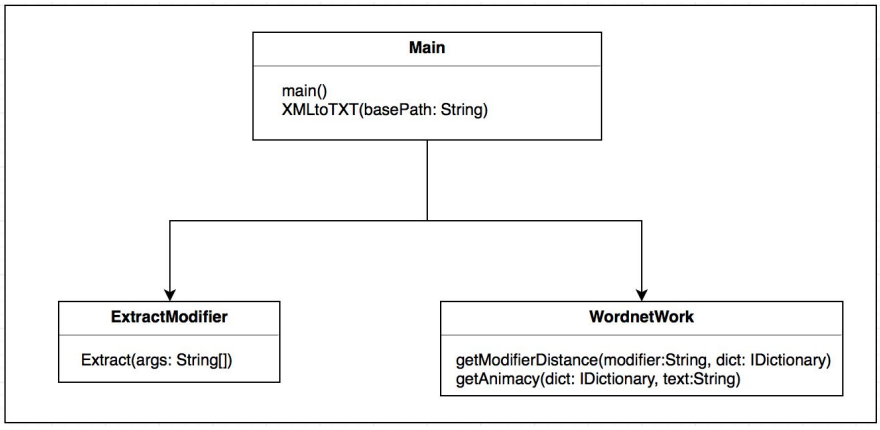

Fig. 20: Class Diagram # 126

User Story #159

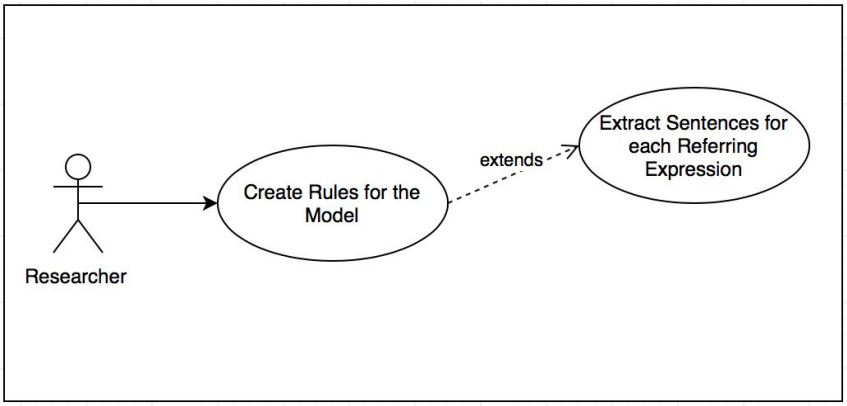

Fig. 21: Use Case Diagram # 159

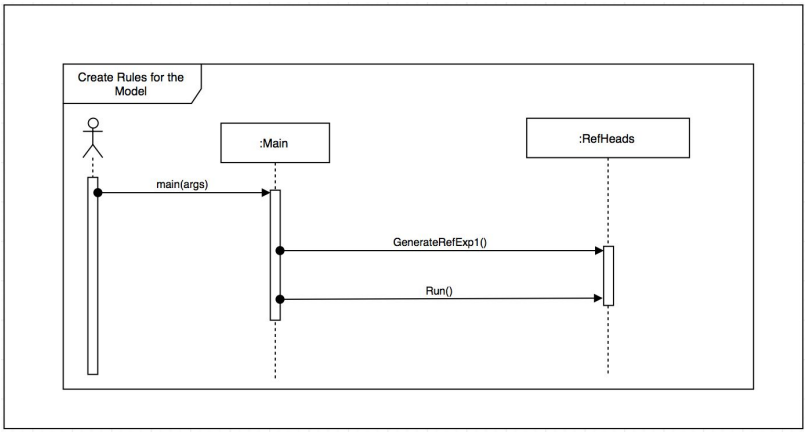

Fig. 22: Sequence Diagram # 159

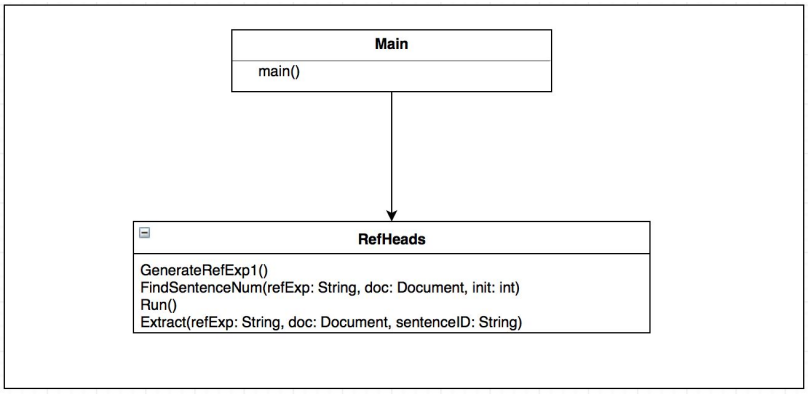

Fig. 23: Class Diagram # 159

Appendix B - Sprint Retrospective Reports

Sprint 1

Date: 1/27
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 3:00pm
End time: 4:00pm

What went wrong?

- Did we do a good job estimating our team's velocity?
  - The velocity could have been greater
- Did we do a good job estimating the points (time required) for each user story?
  - Points should have been greater. Papers and book took a long time to read and document because of the complexity of the material as well as the length of the book.
- Did each team member work as scheduled?
  - Yes

What went right?

- Had a good start to the project related to reading papers and setting up dependency parse.

How to address the issues in the next sprint?

- How to improve the process?
  - Try to finish the use cases and documentation in a more timely manner so that not a lot of time is spent catching up with the documentation
- How to improve the product?
  - Discuss doubts and issues with Dr. Finlayson more

Sprint 2

Date: 2/10
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 3:00pm
End time: 4:00pm

What went wrong?

- Did we do a good job estimating our team's velocity?
  - Yes
- Did we do a good job estimating the points (time required) for each user story?
  - Points should have been greater. User stories and documentation took a long time.
- Did each team member work as scheduled?
  - Yes

What went right?

- Set up the Stanford NLP
- Extracted the dependency parse

- Continued setting up the SRL.

How to address the issues in the next sprint?

- How to improve the process?
  - Break up the tasks inside the use cases better and learn how to use mingle in a more efficient manner
- How to improve the product?
  - Discuss doubts and issues with Dr. Finlayson more

Sprint 3

Date: 2/24
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 3:00pm
End time: 4:00pm

What went wrong?

- Did we do a good job estimating our team's velocity?
  - Yes
- Did we do a good job estimating the points (time required) for each user story?
  - Yes
- Did each team member work as scheduled?
  - Yes

What went right?

- Set up the SRL
- Extracted the semantic parse
- Extracted the needed semantic subjects of the sentences

How to address the issues in the next sprint?

- How to improve the process?
  - Be more organized with planning out the workload so as to not go back and forth between multiple tasks
- How to improve the product?
  - Discuss doubts and issues with Dr. Finlayson more

Sprint 4

Date: 3/10
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 3:00pm
End time: 4:00pm

What went wrong?

- Did we do a good job estimating our team's velocity?
    - Yes
- Did we do a good job estimating the points (time required) for each user story?
    - Yes
- Did each team member work as scheduled?
    - Yes

What went right?

- Annotated more stories
- Extracted the false positive and false negative referring expressions
- Did error analysis on referring expression SVM result

How to address the issues in the next sprint?

- How to improve the process?
    - Be more organized with planning out the workload so as to not go back and forth between multiple tasks
- How to improve the product?
    - Discuss doubts and issues with Dr. Finlayson more

Sprint 5

Date: 3/24
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 3:00pm
End time: 4:00pm

What went wrong?

- Did we do a good job estimating our team's velocity?
    - No
- Did we do a good job estimating the points (time required) for each user story?
    - Yes
- Did each team member work as scheduled?
    - Yes

What went right?

- Calculated modifier distance
- Got animacy information from wordnet sense of the noun

How to address the issues in the next sprint?

- How to improve the process?
    - Be more organized with planning out the workload so as to not go back and forth between multiple tasks
- How to improve the product?
    - Discuss doubts and issues with Dr. Finlayson more

Sprint 6

Date: 4/7
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 3:00pm
End time: 4:00pm

What went wrong?

- Did we do a good job estimating our team's velocity?
  - No
- Did we do a good job estimating the points (time required) for each user story?
  - Yes
- Did each team member work as scheduled?
  - Yes

What went right?

- Started working on the rule-based model
- Extracted the heads of referring expressions

How to address the issues in the next sprint?

- How to improve the process?
  - Be more organized with planning out the workload so as to not go back and forth between multiple tasks
- How to improve the product?
  - Discuss doubts and issues with Dr. Finlayson more

Appendix C - Sprint Review Reports and Daily Scrum

Sprint 1

Date: 1/27
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00pm
End time: 3:00pm

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #107 Reading Animacy Papers + Books

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 2

Date: 2/10
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00pm
End time: 3:00pm

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #125 Set up Stanford NLP + Extract Subjects from Dependency Parse

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 3

Date: 2/24
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00pm
End time: 3:00pm

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #130 Set up SRL and Extract ARG0

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- User Story # 126 Extract Modifier Animacy Properties from WordNet

- How this should be reflected on the user story definition in Mingle:
    - This story was longer than expected and had to be pushed to be completed in the next few sprints ie Sprint 4 or Sprint 5

Sprint 4

Date: 3/10
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00pm
End time: 3:00pm

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #151 Annotate more Stories + Error Analysis on Referring Expressions

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 5

Date: 3/24
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00pm
End time: 3:00pm

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #126 Extract Modifier Animacy Properties from WordNet

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 6

Date: 4/7
Attendees: Geeticka Chauhan, Labiba Jahan, Mark Finlayson
Start time: 2:00pm
End time: 3:00pm

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #159 Create Rules for the Model

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Daily Scrum Meeting Minutes:

Sprint 1

Date: 1/16
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. This was the first scrum meeting
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Set up project environment in mingle
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 1/17
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Set up project environment in mingle.
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Read papers and books on animacy
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 1/18
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Read papers and books on animacy
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Set up Stanford NLP and read on it
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

2. Maven was giving issues with the POM.xml, but adding extra jar files to the classpath solved it.

Date: 1/19
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Set up Stanford NLP and read on it
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start setting up dependency parse
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Parsing the XML files.

Date: 1/20
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Started setting up dependency parse
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Get the grammatical subjects from dependency parse per sentence
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 1/23
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Get the grammatical subjects from dependency parse per sentence
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Get the grammatical subjects from dependency parse in the overall document
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. The LinkedHashTable gave issues

Date: 1/24
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Get the grammatical subjects from dependency parse in the overall document
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start annotating 7 stories by word animacy
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Annotating multiple stories is hard on the eyes and can be tough.

Date: 1/25
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Start annotating 7 stories by word animacy
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Continue and finish annotating the 7 stories
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 1/26
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Continue and finish annotating the 7 stories
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Plan for Sprint Retrospective and Review
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 1/27
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Sprint Retrospective and Review
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Summarize papers + Crete Google Drive for notes
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Sprint 2

Date: 1/30
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Summarize papers + Crete Google Drive for notes
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Sprint Planning + team meeting
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 1/31
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Sprint Planning + team meeting
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Read about the Skip-gram model in word to vec representation
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/1
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Read about the Skip-gram model in word to vec representation
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Meet Josh to understand SRL system and set it up
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/2
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Meet Josh to understand SRL system and set it up
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start parsing XML files to extract sentences
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/3
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Start parsing XML files to extract sentences
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Finish parsing to get ARG0
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/6
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Start parsing XML files to extract sentences
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Set up private GitHub repo for RefAnimacy for Labiba + figure out how to switch from Maven
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Over 200 dependencies so almost impossible to switch from Maven - must package them all into one jar

Date: 2/7
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Set up private GitHub repo for RefAnimacy for Labiba + figure out how to switch from Maven
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Fix basePath issues in GitHub repo
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Code gives a NullPointerException - must figure it out

Date: 2/8
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Fix basePath issues in GitHub repo
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Read up on WordNet distances between modifiers + extract modifiers - may need to talk to Dr. Finlayson about this
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

2. Still must figure out NullPointerException

Date: 2/9
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Read up on WordNet distances between modifiers
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Work on documentation related to senior project
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/10
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Work on documentation related to senior project
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Extract modifiers + Figure out how to calculate distances between modifiers
    3. Complete senior project documenation
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Senior Project documentation is longer than expected

Sprint 3

Date: 2/13
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Read up on JWS similarity calculator and calculate Fa and Fm values to use as feature in SVM for coreference chain animacy.
- What is planned to be done until the next scrum meeting?

1. Finish product backlog
2. Figure out the issue with Fa and Fm padding calculation
3. Implement SVM for coreference chain animacy

- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. The padding for Fa and Fm is not working - somehow not able to extract the Fa and Fm of the last referring expression in the coreference chain

Date: 2/14
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Figured out how to do the padding
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Move on to the modifier issue because the SVM issue is related to indexing which was worked on by Labiba
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. The Fm values are still being returned as 0's for some chains, which should not be happening

Date: 2/15
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Extracted the modifiers from the dependency parse and read up on the JWI library
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Figure out how to extract related modifiers from wordnet and how to calculate distance from a modifier to an animate noun (basically how to characterize modifier by animacy)
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. JWI library is unfamiliar so there is a large learning curve

Date: 2/16
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Played around with JWI to figure out how to get related modifiers
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. How to calculate distance from a modifier to an animate noun (basically how to characterize modifier by animacy)
    3. Set up JWS library
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. JWI library is unfamiliar so there is a large learning curve and the concept behind calculating distance from modifier to noun is unfamiliar to me

Date: 2/17
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Set up JWS library + got started on distance calculation between two words
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Figure out how to do distance calculation
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Making WordNet work was a problem because JWI and JWS wasn't recognizing it as a directory at first + had to download an additional WordNet::Similarity document to work with JWS code

Date: 2/20
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Still working on distance calculation between two words
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Figure out the error with distance calculation

- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Words like apple and banana are being recognized by JWS, but others are not such as alive and even though they do exist in wordnet, JWS says that they do not - must troubleshoot this issue

Date: 2/21
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Read through Wordnet to see why the distance between adjectives is not calculated
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Figure out the error with distance calculation
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/22
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Discuss with Labiba about why the False Negatives are being generated
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Figure out how to reduce the False Negative rate
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/23
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Try more alternatives inside the code to find out which distances are possible to calculate
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Decide what kinds of distances are being calculated and which aren't
- What are the hurdles?

1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/24
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Decided that in general, noun distances were being calculated but adjective ones were not
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Figure out the Null Pointer error with distance calculation
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Sprint 4

Date: 2/27
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Still trying to figure out the distances issue, got 30 more stories
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start annotating the stories
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 2/28
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Distances issue figuring out
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start annotating the stories
- What are the hurdles?

1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/1
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Distances issues solved
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start annotating the stories
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/2
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Annotated Story 16, 17 and 18
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Continue annotating the stories
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/3
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Annotated Story 18, 19, 20, 21
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Continue annotating the stories
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/6

Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Annotate Story 22, 23, 24
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Continue annotating
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/7
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Annotate Story 25, 26
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Extract the False Positive and False Negative referring expressions
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/8
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Extract the false positive and false negative referring expression and do error analysis
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start calculating the distance between the modifiers from the Stories and “aliveness”
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Code takes at least 15 minutes to run, so each little change must not break the code.

Date: 3/9
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Figuring out the best way to calculate modifier distances
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Calculate modifier distances and feed to the SVM
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/10
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Calculate modifier distances to feed to SVM
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Fix projects to make them mobile
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Sprint 5

Date: 3/13
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Sprint Planning and Fix Projects to make them mobile
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Parsing referring expressions with sentence numbers from workbench
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/14

Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Start parsing referring expressions with sentence numbers
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Finish parsing
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/15
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Finish parsing
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Get heads of referring expressions
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/16
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Start working on getting heads of referring exp
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Finish getting the heads
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/17
Attendees: Geeticka Chauhan, Labiba Jahan

Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Finish getting the heads
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Fix the sentence numbers according to stanford NLP
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features
    2. Sentences are numbered differently in Story Workbench and Stanford NLP

Date: 3/20
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Work on fixing the sentence numbers
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Finish fixing the sentence numbers
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/21
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Finish fixing the sentence numbers
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Get animacy from wordnet sense
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/22
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Get animacy from wordnet sense
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start on research paper for VIP project
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/23
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Start on research paper for VIP project
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Documentation
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/24
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Documenation + Sprint Retrospective and Review
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start annotating newer coref chains in stories
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Sprint 6

Date: 3/27
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Sprint Planning + Work on VIP Paper
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. VIP Presentation for FIU Undergrad Conference
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/28
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. VIP Presentation for FIU Undergrad Conference
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Present for VIP Presentation and practice
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/29
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Present for VIP Presentation and practice
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Start annotating stories 34-46 for animacy due to new coreference chains
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/30
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Start annotating stories 34-46 for animacy due to new coreference chains
- What is planned to be done until the next scrum meeting?
    3. Finish product backlog
    4. Work on VIP Paper, Continue annotating
- What are the hurdles?
    2. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 3/31
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Work on VIP Paper
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Continue working on paper and annotations
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 4/3
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Annotate stories 34-46
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Continue annotating stories 34-46
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 4/4
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Annotate stories 34-46
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Finish annotating stories 34-46
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 4/5
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Finish annotating stories 34-46
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Continue working on VIP Paper
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 4/6
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Continue working on VIP Paper
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. Finalize the rules + Sprint Review and Retrospective + Documentation
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Date: 4/7
Attendees: Geeticka Chauhan, Labiba Jahan
Start time: 9:00PM
End time: 9:30PM

Student1: Geeticka Chauhan

- What was done since the last scrum meeting?
    1. Finalize the rules + Sprint Review and Retrospective + Documentation
- What is planned to be done until the next scrum meeting?
    1. Finish product backlog
    2. VIP Paper finish and finalize
- What are the hurdles?
    1. Waiting on Dr. Finlayson to send us the rest of the product features

Appendix D - Installation/Maintenance, Future Work and other documents

Installation and Maintenance Information

Following is the installation information related to the code:

(No screenshots needed as installation is straightforward)

1. In Eclipse IDE, go to File -> Import -> Existing Java Projects
2. Import all 3 projects, one by one
3. Run, by following directions in comments written in code. For example, the code will indicate different steps to follow and you simply have to comment out the correct lines and the code will run

Future Work

Following is the ongoing work in the system and the future work that is to be performed:

1. Ongoing: Rule-based system implementation to improve the result of Coreference Chains.
2. Automatically extracting character names and get their roles using animacy information

Summary on Important Papers for Animacy Detection

1. NP Animacy Identification for Anaphora Resolution

Constantin Orăsan

Richard Evans

Use rule-based and ML method (classification) in order to find the animacy of Noun Phrases. They use WordNet - first classify different senses as animate or not and on the basis of that perform ML on heads of NPs to find out the animacy of the whole NP. The do intrinsic evaluation (standard F1 etc) but do extrinsic evaluation of animacy detection by checking how well it performs under anaphora resolution (MARS system) and word sense disambiguation. This is similar to what we want to do except we want to use animacy to improve coref chains and even then this is only a part of our research because they don't use coreference information to determine animacy.

Related from same authors:

★ Improving anaphora resolution by identifying animate entities in texts

Simply gives a detailed explanation of what the above paper implements.

★ Learning to identify animate reference**s**

Describes the machine learning method that was used in original paper.

2. A Machine Learning Approach to Anaphora Resolution Including Named Entity Recognition, PP Attachment Disambiguation, and Animacy Detection (thesis)

Anders Nøklestad

Similar to #1, it uses animacy etc to be able to test them on the basis of whether they improve anaphora resolution.

3. Animacy Detection in Stories

Folgert Karsdorp Marten van der Meulen

Theo Meder Antal van den Bosch

Implements word animacy in order to extract the characters from a story. The features they use (with a window of 3) are Lexical Features (word forms and lemma), syntactic features (dependency parse to check which word is nsubj or nobj), morphological features (POS tags), semantic features (word embedding using skip-gram model to vectorize each word - purpose is "similarities between animate words can be estimated by inspecting the context in which they occur.") The best results are gotten by just using the words + POS + embedding feature. Standard evaluation done.

4. Morphology of the Folktale

Propp

This describes Propp's theories about the way Russian folktales are structured. Propp identified main themes between these folktales as well as the structure, such as background, introduction of characters, body, etc. This helps in understanding the way the data is structured and will help in identifying the characters and later classifying them.

Error Analysis

Main categories of False Positive (Those that should be inanimate but are marked as animate)

Time of day (morning, midnight, dawn) ----- NER

Mace (a club) and hut ---- Sense work

It, here

Main categories of False Negative (Those that should be animate but are not marked as animate)

Pronouns excluding it, including possessive ----NER or POS?

Proper Nouns ----NER

Animals ---- Sense

Humans (such as daughters and sisters) ---- Sense

Might be possible to solve using:

Find the head of referring expression first

Run named entity recogition on that head or figure out the wordnet sense and whether it an "animate being" or "living being" in wordnet (ie check out hyponymy because it correlates with "type of" but to go up the hierarchy must use hypernym)

But sense of the word shouldn't be enough to rule it out as animate or inanimate- modifiers of the head need to be looked at and then see if they give an indication of animacy or not

Idea is to have a hybrid rule-based and statistical system so that we can use both techniques to be able to accurately identify animacy

Draft Paper

Starting from the next page, the draft paper pages have been attached, which act as a submission of the Vertically Integrated Projects (VIP) side of the senior project.

Multi-Sieve approach to Character Identification and Classification

Geeticka Chauhan*
Florida International University
Miami, FL, USA
gchau005@fiu.edu

Labiba Jahan*
Florida International University
Miami, FL, USA
ljaha002@fiu.edu

Mark A. Finlayson
Florida International University
Miami, FL, USA
markaf@fiu.edu

Abstract

Our aim is to perform automatic classification of characters in folktales by using the concept of animacy. We define animacy as the property of being alive and communicative, for example: people and animals. In addition to these entities, folktales often involve characters that are not seen as animate in the real world, and this makes our problem a challenging one. Unlike other work, we attempt to classify the animacy of coreference chains using a multi-sieve approach by combining a rule-based method and classifier. Our results are promising so far, and reaching very close in F1 measure of the state of the art techniques.

1 What is animacy?

We define animacy of coreference chains by their constituent referring expressions. A referring expression is animate if its main subject is animate (i.e. alive and communicative). We follow the rules of animacy defined by (Quirk et al., 1985), which defines animate entities as personal and non-personal noun phrases. Personal noun phrases are classified by male, female, dual, common and collective, whereas non-personal noun phrases are classified by common, collective, higher or lower nouns. In short, we include titles such as daughter, doctor, baby, family (collective), animals (but not insects) as well as talking inanimate entities such as talking trees (due to the fairytale nature of folktales).

2 Background

There have been several approaches to and uses of animacy detection in stories. (Orăsan and Evans, 2007), (Evans and Orăsan, 2000) as well as (Orăsan and Evans, 2001) explain a WordNet and Machine Learning (ML) based method to animacy detection applied to the problems of Word-Sense Disambiguation (WSD) as well as Anaphora Resolution (AR). Another work that utilizes animacy detection, along with other techniques such as Named Entity Recognition (NER) to improve the performance on the task of AR is (Nøklestad, 2009).

Another method for animacy detection by (Bloem et al., 2013) involve the use of the k-nearest neighbor algorithm using lexical features such as if the noun appears as the subject to the verb "to think." (Bowman and Chopra, 2012) take the animacy detection problem further and use a maximum entropy classifier to classify noun phrases by classes such as human, vehicle, time, animal etc.

Aside from animacy detection, other methods involving encoding character information into ontologies have been proposed by (Declerck et al., 2016) for the purpose of classifying characters in folktales. (Karsdorp et al., 2012) use the intentionality of subject i.e. direct and indirect speech in order to extract the cast of a story as well as rank it by its importance.

The machine learning methods we use in our work are closely related with those of (Karsdorp et al., 2015). They implement a Support Vector Machine (SVM) to classify words according to their animacy and receive good results based on it.

3 Experimental Setup

In this section we explain our problem, the data that we use and the method we use to solve this problem. Our model is explained in Figure 1.

3.1 Problem

We are classifying coreference chains by their animacy by using a Support Vector Machine (SVM) classifier and a rule-based method, utilizing the concepts of NER, WordNet sense, Part of Speech

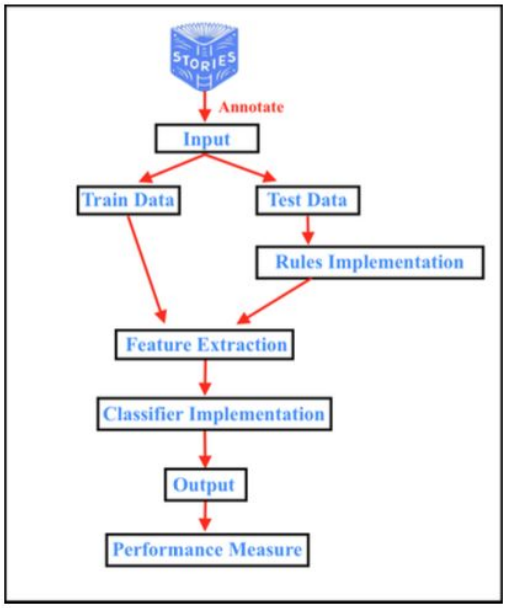

Figure 1: Model of our multi-sieve approach

(POS) and devising rules for text in quotation marks on the referring expressions. Based on the classification of the referring expressions, we use majority-vote to classify the coreference chains they belong to.

3.2 Data and Annotation

We make use of 15 Russian Folktales from (Finlayson, 2015) that have been annotated with the Natural Language Processing (NLP) pipeline such as sentences, parts of speech etc. Additionally, we performed hand-annotation of the coreference chains based on their animacy, and example of which is provided by Figure 2. We are working on extending our models to 46 stories in order to get better results.

3.3 Rule-Based Method

The rule-based method we follow is in the process of being implemented. We extract the heads of every referring expression by extracting the overall governor in the dependency parse. For each of these heads, we look at POS, NER, WordNet sense (as well as that of its modifier) and rules for quotes:

- **POS**: we mark non-possessive pronouns such as "he" and "she" as animate, but not "it."
- **NER**: we utilize the PERSON label of the heads to mark as animate
- **WordNet sense**: we utilize the subclass and superclass relationship between "living_being" and the head to decide on the animacy. Additionally, we look at the sense of the modifiers to decide whether they change the animacy of the head.
- **Rules for Quotes**: working on deciding the rules for referring expressions that consist of quotes.

3.4 Machine-Learning Method

We use a Support Vector Machine (SVM) classifier and the skip-gram model introduced in (Mikolov et al., 2013) in order to classify the referring expressions by animacy.

3.4.1 Evaluation

We divide our initial data of 15 stories into 10 for training and 5 for test. We use the standard measures of Precision, Recall, Accuracy and F1 in order to evaluate our results, defined by (Powers, 2011). Precision refers to the fraction of actual animate referring expressions out of all the ones classified as animate. Recall refers to the fraction of all actual animate referring expressions that are classified as animate. Accuracy refers to the proportion of true results in terms of correctly marked animate and non-animate among the total classified. Finally, F1 measure is the harmonic mean of precision and recall, to combine both measures. Relevant concepts for these measures are True and False Positives (TP/FP), which mean the number of predicated animate that were correct/incorrect and True and False Negatives (TN/FN), which mean the number of not-predicted animate that were correct/incorrect.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

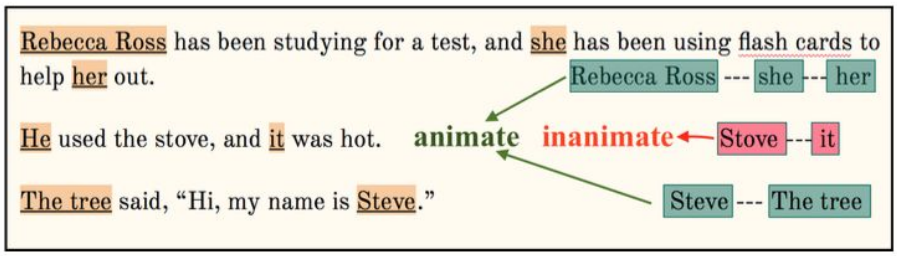

Figure 2: Sentences with their animate and inanimate chains

3.4.2 Features

The features for the model are binary (yes/no) and are word, POS, word embeddings, dependency head, Semantic Role Labeler (SRL) head, coreference. They are explained as follows:

- **Word**: 3 words before and after the concerning words.
- **POS**: (word animacy model) Part of Speech of the concerning word such as noun, adjective etc.
- **Word Embedding**: (word animacy model) Word Embedding generated by Word2Vec's skip-gram architecture.
- **Coreference**: (word animacy model) Whether the current word belongs to a coreference chain.
- **Dependency Head**: Grammatical subject of each sentence according to dependency parse.
- **SRL Head**: Semantic subject of each sentence according to the semantic role labeler.

3.4.3 Models

First, we built a model similar to (Karsdorp et al., 2015) for word animacy in order to get a baseline measure. We used the word, POS, word embedding and coreference features for this.

The second model we built was making animacy predictions for referring expressions and used word, POS (just looking at noun and pronoun) and Dependency Head features.

The third model we built made predictions for referring expressions as well and used word and SRL head features. Our last model was not based upon machine learning, but took a majority vote of the predictions of referring expressions from the second model to make predictions about animacy of coreference chains.

4 Technologies Used

As input, we used the annotated versions produced by (Finlayson, 2011) i.e. the MIT Story Workbench. For feature extraction, we made use of the (Deeplearning4j Development Team, 2017) i.e. the Word2Vec library in Java, the MIT Java Wordnet Interface from (Finlayson, 2014), the Semantic Role Labeler (SRL) from (Finlayson, 2015) and the Stanford CoreNLP from (Manning et al., 2014).

5 Results

The Results are summarized by Figure 3 and we find that the model 3 using word + SRL head features gives us the best results.

The first model was just used to provide a good baseline for our data, in order to evaluate how well we did with our models pertaining to referring expressions.

The second model provided a start to our referring expression classifier and worked really well due to our dependency head feature. We hypothesized that if we looked at semantic subjects as opposed to grammatical subjects (which is what the dependency parse was giving us), we would be able to get the "main thing" being talked about in the sentence and that most of the time, this "thing" would be alive if it was the head of a sentence. This lead us to creating the third model involving only the word and SRL head features. The reason that we eliminated the additional features was because their effect on the prediction was close to none after we included the SRL head feature.

The fourth model is based on the third one - we took the prediction of the majority of referring expressions to assign animacy to each coreference chain. Because this model does not give satisfactory results, we are working on error analysis and extending our third model to 46 stories instead of just the original 15. The preliminary error analysis

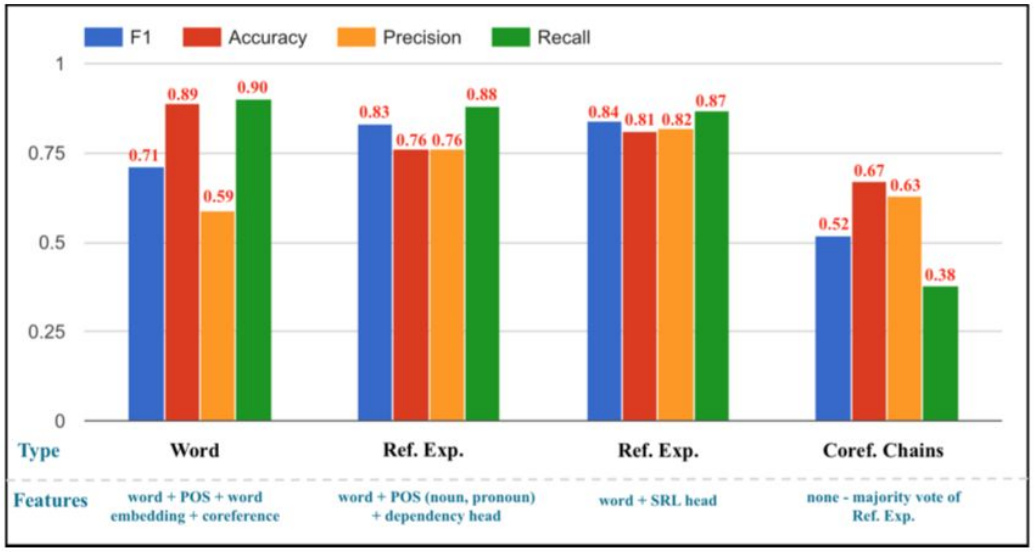

Figure 3: Results

revealed the need for the rule-based system, which we are working to perfect.

6 Conclusion

Our work provides several contributions. Firstly, we created clean annotated data to work well with our SVM classifier. Secondly, we created an experimental set-up for animacy classification of coreference chains, which we hypothesize will be better at character extraction than word animacy classification work done previously. Thirdly, we implemented an SVM classifier for animacy of referring expressions to provide a lead into the animacy of coreference chains. And finally, we were able to get good results of 84% F1 score for the animacy of referring expressions.

7 Future Work

We are currently working more on the rule-based model and plan to extend our word + SRL head model to all 46 stories in our corpora. This involves annotation of the chains within those stories, re-checking those annotations and extending our model to work on those. Once we are able to obtain a more satisfactory result for the F1 score of the model using rules and machine learning, we will be able to move on to character name extraction and then classification by roles such as hero and villain.

Acknowledgments

This was joint work with Labiba Jahan, a Ph.D. student working under Dr. Mark Finlayson in the Cognac Lab at Florida International University.

References

Jelke Bloem, Gosse Bouma, et al. 2013. Automatic animacy classification for dutch. *Computational Linguistics in the Netherlands Journal*, 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: Student Research Workshop*, pages 7–10. Association for Computational Linguistics.

Thierry Declerck, Tyler Klement, and Antonia Kostova. 2016. Towards a wordnet based classification of actors in folktales. In Verginica Barbu Mititelu, Corina Forăscu, Christiane Fellbaum, and Piek Vossen, editors, *Proceedings of the Eighth Global WordNet Conference*. Global WordNet Association, GWA, 1.

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. Citeseer.

Mark A Finlayson. 2011. The story workbench: An extensible semi-automatic text annotation tool. In *Intelligent Narrative Technologies*.

Mark Alan Finlayson. 2014. Java libraries for accessing the princeton wordnet: Comparison and evaluation. In *Proceedings of the 7th Global Wordnet Conference, Tartu, Estonia.*

Mark A Finlayson. 2015. Propplearner: Deeply annotating a corpus of russian folktales to enable the machine learning of a russian formalist theory. *Digital Scholarship in the Humanities*, page fqv067.

Folgert Karsdorp, Peter Van Kranenburg, Theo Meder, and Antal Van den Bosch. 2012. Casting a spell: Identification and ranking of actors in folktales. In *Proceedings of the Second Workshop on Annotation of Corpora for Research in the Humanities (ACRH-2)*, pages 39–50. Citeseer.

Folgert Karsdorp, M Meulen, Theo Meder, and APJ van den Bosch. 2015. Animacy detection in stories.

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Anders Nøklestad. 2009. A machine learning approach to anaphora resolution including named entity recognition, pp attachment disambiguation, and animacy detection.

Constantin Orăsan and Richard Evans. 2001. Learning to identify animate references. In *Proceedings of the 2001 workshop on Computational Natural Language Learning-Volume 7*, page 16. Association for Computational Linguistics.

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

David Martin Powers. 2011. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation.

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

References & Acknowledgement

Please refer to the draft paper in Appendix D for references

This project was joint work with Labiba Jahan, Ph.D. candidate at Cognac Lab of Florida International University. I would like to thank her for being my partner as well as Dr. Mark Finlayson for mentoring our project and meeting with us weekly.