

20

Iterative Design and Natural Language Processing in the WISE Project

by

Aalok Bipin Shah

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degrees of
Bachelor of Science in Computer Science and Engineering
and Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

May 21, 1999

[June 1999]

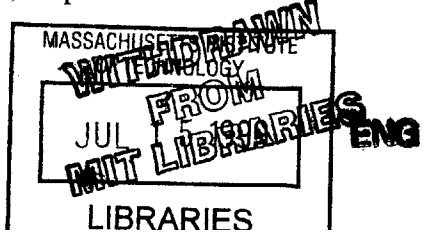
© Copyright 1999 Aalok Bipin Shah. All rights reserved.

The author hereby grants to M.I.T. permission to reproduce and
distribute publicly paper and electronic copies of this thesis
and to grant others the right to do so.

Author _____
Department of Electrical Engineering and Computer Science
May 21, 1999

Certified by _____
Justine Cassell
AT&T Career Development Assistant Professor of Media Arts and Sciences
Program in Media Arts and Sciences
Thesis Supervisor

Accepted by _____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses



Iterative Design and Natural Language in the WISE Project
by
Aalok Bipin Shah

Submitted to the Department of Electrical Engineering and Computer Science

May 21, 1999

In Partial Fulfillment of the Requirements for the Degree of
Bachelor of Science in Computer Science and Engineering
and Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

The SAGE system was built to fill the need for “technological tools ... designed to encourage self-reflection” (Bers 1997). Bers chose to approach this through personal conversational storytelling. Due to limitations of SAGE, a web-based successor, WISE, was built. It continued to fill SAGE’s role as an environment for conversational storytelling, as well as a construction kit for new storytelling characters with their own stories.

This thesis addresses some of the limitations of the original version of WISE, and some approaches to solutions to those limitations. The technical implementation of WISE and the results of experiments run to test modifications to the story matching algorithm are presented. Those modifications include the addition of verbs automatically extracted from stories to story keywords specified by the user, as well as the heuristic stemming of keywords. The original version of the algorithm used heuristic inflection rather than stemming. Results show that automatic verb extraction as a supplement to user-chosen keywords results in better story matches than keywords alone. They also indicate that heuristic inflection of keywords gives better results than heuristic stemming.

Thesis Supervisor: Justine Cassell

Title: AT&T Career Development Assistant Professor of Media Arts and Sciences

Table of Contents

Abstract	2
1 Introduction	6
1.1 Overview of Thesis	7
2 Interaction and Character Design	7
2.1 Character Interaction	8
2.2 Character Creation	11
2.3 Character Editability	16
3 Goals and Context	18
3.1 SAGE	18
3.2 WISE: a Successor to SAGE	19
4 Literature review	20
4.1 WISE and the User	20
4.2 The Technological Basis of WISE	22
5 The Limitations of WISE	24
5.1 Loss of Iterative Design	24
5.2 Loss of Context	25
6 Solutions	26
6.1 Regaining Iterative Design	26
6.2 Processing of Stories	28
6.2.1 Verb Extraction	29
6.2.2 Stemming	30
7 Technical Design and Implementation of WISE	31

7.1	Design Decisions	31
7.2	Implementation of WISE	32
7.2.1	Tools Used in WISE	32
7.2.1.1	HTML	33
7.2.1.2	Perl	33
7.2.1.3	WordNet	34
7.2.1.4	The Constraint Grammar Parser of English	35
7.2.1.5	The Porter Stemmer	35
7.2.2	Compilation	36
7.2.2.1	WordNet Expansion	36
7.2.2.2	Verb Extraction	38
7.2.2.3	Culling Stopwords	39
7.2.2.4	Stemming	39
7.2.2.5	Keyword Sorting and Collection	40
7.2.3	Interaction	41
7.2.3.1	The Story Matching Algorithm	41
8	Experimentation and Results	42
8.1	Methodology	43
8.2	First Survey and Results	44
8.3	Second Survey and Results	46
9	Evaluation	49
9.1	Heuristic Stemming	49
9.2	Verb Extraction	51

10	Conclusion	53
11	Future Work	54
12	References	55
Appendix A	The First Survey	58
Appendix B	The Second Survey	66
Appendix C	Compilation Results	75
Appendix D	Story Matching Results	81

1 Introduction

Bers (1997) built the Storyteller Agent Generation Environment (SAGE) to address a need for “technological tools specifically designed to encourage self-reflection.” SAGE is not, however, very accessible. Sharing characters with others, which allowed creators to find and fix bugs in their characters, is non-trivial since characters are tied to a single machine. Programming a conversational interface for a character requires the use of a LISP-like language, which children have found non-intuitive. These drawbacks were addressed in the Web Interactive Storytelling Environment (WISE), a web-based successor to SAGE. WISE’s focus became storytelling, with less attention paid to the conversational aspect of interaction with a character. While WISE is easy to use, the first implementation has its limitations. Foremost among those are the lack of the iterative design (the cycle of testing and improving characters) prized in SAGE and the weakening of characters’ socio-cultural context, which in SAGE made characters emotionally believable.

These two deficiencies in WISE will be addressed in this thesis. Character editability has been given to WISE to restore SAGE’s cycle of iterative design, with its focus shifted from conversation to storytelling. At the same time, story matching has been addressed in the hope that intelligent story matching can replace SAGE characters’ intelligent conversation in the establishment of socio-cultural context and emotional believability.

1.1 Overview of Thesis

Chapter 2 introduces WISE by describing character interaction and creation. Chapter 3 presents the immediate context in which WISE exists and is being improved, as well as the goals of WISE and its predecessor, SAGE. Chapter 4 presents the larger context, both technical and humanistic, surrounding this project. Chapter 5 discusses in detail the limitations of the first implementation of WISE. Chapter 6 proposes solutions that address those limitations. In Chapter 7, the technical design and implementation of WISE is provided. Chapter 8 presents the evaluation of the improvements in story matching and the results. In Chapter 9, the results presented in Chapter 8 are discussed and justified. In Chapter 10, the issues addressed in this thesis are summarized and concluded. Chapter 11 points out future directions for the testing and development of WISE. Chapter 11 is followed by a bibliography and appendices.

2 Interaction and Character Design

WISE consists of two distinct phases: interacting with an existing storytelling character, and creating new one. In the interaction phase, a user has a short conversation with the character, in which the character takes the conversational initiative in evoking information, personal experiences, and stories from the user. At the end of the interaction, the character chooses the story in its collection that is most relevant to the user's responses.

In the creation phase, the character creator determines several aesthetic aspects of the character, scripts the character's conversation, and adds stories and story keywords to the character's story database. Character editing, which is very similar to creation, allows

creators to change their characters' aesthetics and conversation, as well as add or change stories and keywords.

2.1 Character Interaction

A user invokes a character by simply going to its web address using an ordinary web browser. From there, interaction consists of a conversation between the character and the user. In each segment of the conversation, the user reads text displayed on the character's page and responds appropriately by typing into a text input box (see Figure 2-1). The character's side of the conversation is written during the creation process. Thus, the conversation is linear and, on the character's side, scripted. There may be between one and four conversational exchanges during the course of the conversation. In the last exchange, the character may evoke a story from the user. At the end of the conversation, the user's accumulated responses are processed by the story matching algorithm.

The story matching algorithm (described below) takes the user's responses to the character's prompts (accumulated by the character), compares them to the character's match list of story keywords, and returns the story that best matches them (see Figure 2-2).

At the end of an interaction, the user has the option to start another interaction by returning to the character's first page.



AESOP

(A character created in WISE)

It's good to talk with you. The people who come to me always are thinking about rough times. I'm no stranger to adversity. I was a slave once. I had nothing, not even my freedom. But still I sought to learn and improve myself. And, I served my master as best I could. I began to learn from the actions of animals and people, and from the teachings of the wise. From these I put together stories. The stories I have told made me famous and respected. Even my master said he learned from them, and in gratitude he set me free. Now, I live a good life.

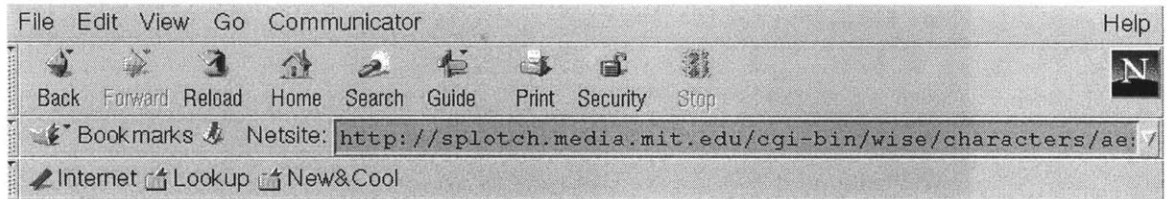
But I am rambling again. Why don't you tell me a story? Tell me about some time when you overcame adversity.

The water was cold and clear. I had my snorkeling gear ready to go. It didn't look too difficult, but then again, looks could be deceiving. I dove into the water, not knowing what to expect. The current was strong, so I had to make an effort to stay in one general area near the shore. When I finally looked through the snorkeling mask into the water, I could not believe what I saw. It was like watching an episode from Jacques Cousteau. The water was crystal clear and the my breathing was controlled. Even

Tell Aesop



Figure 2-1: Aesop evoking a story



AESOP

(A character created in WISE)

I always love to hear about people's experiences. Your story reminded me of one I know. Let me share it with you.

The Fox and the Mask

A Fox had by some means got into the store-room of a theatre. Suddenly he observed a face glaring down on him and began to be very frightened; but looking more closely he found it was only a Mask such as actors use to put over their face. "Ah," said the Fox, "you look very fine; it is a pity you have not got any brains."

Outside show is a poor substitute for inner worth.

I hope we both learn something from the stories we shared. It was nice to talk to you.

[Another conversation with Aesop](#)



Figure 2-2: Aesop returning a story

2.2 Character Creation

Before a character can be created, space for the character must be prepared by a WISE administrator. This allows a measure of control over the number and propriety of the characters in the system. Once space for a character has been prepared, a character creator may begin to work on her character.

As with interaction, the creation process consists of a series of HTML pages and Perl scripts. Each page asks the creator to specify certain aspects of her character and then takes her to the next HTML page.

The first part of creation (i.e. the first page) allows the creator to specify necessary elements and aesthetic aspects of her character (see Figure 2-3). The creator is asked to enter the name of the character, the URL (location on the Internet) of a picture to associate with the character, a color, and the number of conversational exchanges. The name is decided ahead of time, before the WISE administrator prepares space for the character. It is requested here so that the newly created character can be stored in its proper place. The picture that the given URL points to will be displayed at the top of every page during the interaction process, while the color the creator selects will be used as the background color in those pages. Finally, the number of exchanges the user selects will be the number of exchanges before the character returns a story.

The next page in the creation process asks the creator to script the character's side of the interaction conversation (see Figure 2-4). This page is constructed from the values specified in the previous page, so that the creator can get an idea of what her finished character will look like. The conversation scripting page consists of a number of template pages, laid end to end, that correspond to the character's interaction pages. Each

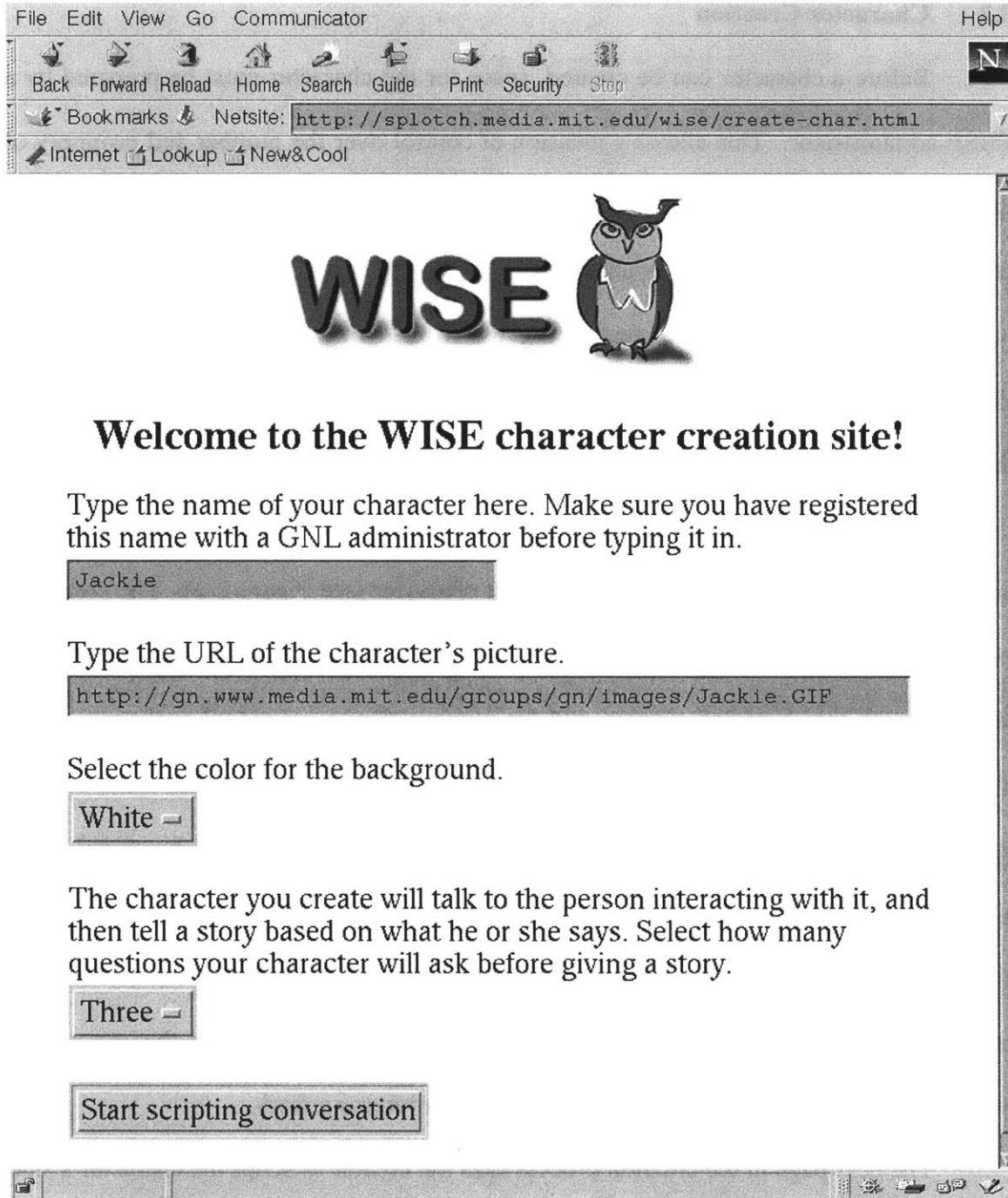


Figure 2-3: Character creation: Aesthetics and conversation turns

of the template pages is an interaction page, including the character's picture and background color, with a text input box in place of the character's text. The creator

scripts the character's side of the conversation by typing into these input boxes. The creator's job in this stage of the process is to write prompts that will involve the users who interacts with this character, and evoke long, complete answers (especially stories) from them.

Once the conversation scripting page is completed, the creator is taken to the story input page (see Figure 2-5). At this point, the creator is finished creating the character proper, and must now add stories and their keywords to the character's database. The story input page has another text input box for the story itself and several lines in which to input keywords and their associated information. The creator must think of up to ten keywords for each story that capture what she thinks is the central meaning of that story. For each keyword, the creator must also specify the word's part of speech, importance, and sense. The importance of a keyword is a number between one and ten (ten being the highest) indicating how significant the associated keyword is to a story. In "The Boy Who Cried Wolf," for example, a keyword like "lie" might get a very high weight, while a word like "sheep" (which has little to do with the moral of the story) may get a much lower weight. The word sense that the creator enters is the single meaning, out of the keyword's many possible meanings, that most closely describes the concept that the creator wants to capture. A creator can use the WordNet lexical database (which has been given a web interface as part of WISE) to view the senses of a word, so that she can select the appropriate sense. The senses of each word in WordNet are numbered, so specifying a sense requires only the word, its part of speech, and a number. The keywords

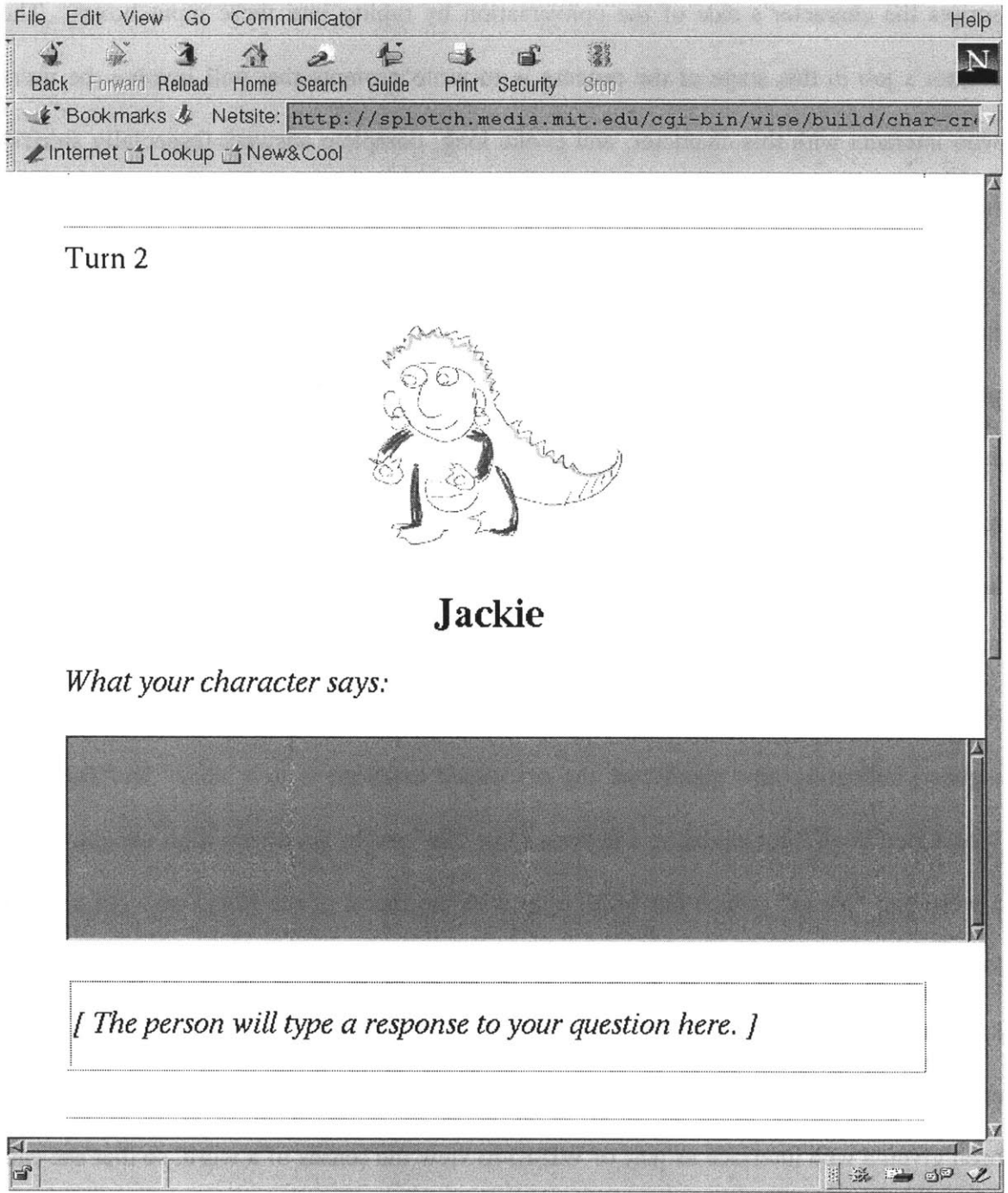


Figure 2-4: Character creation: Conversation

specified here are used in the character's compilation, and become the basis for the match list used by the matching algorithm to find relevant stories during the interaction phase.

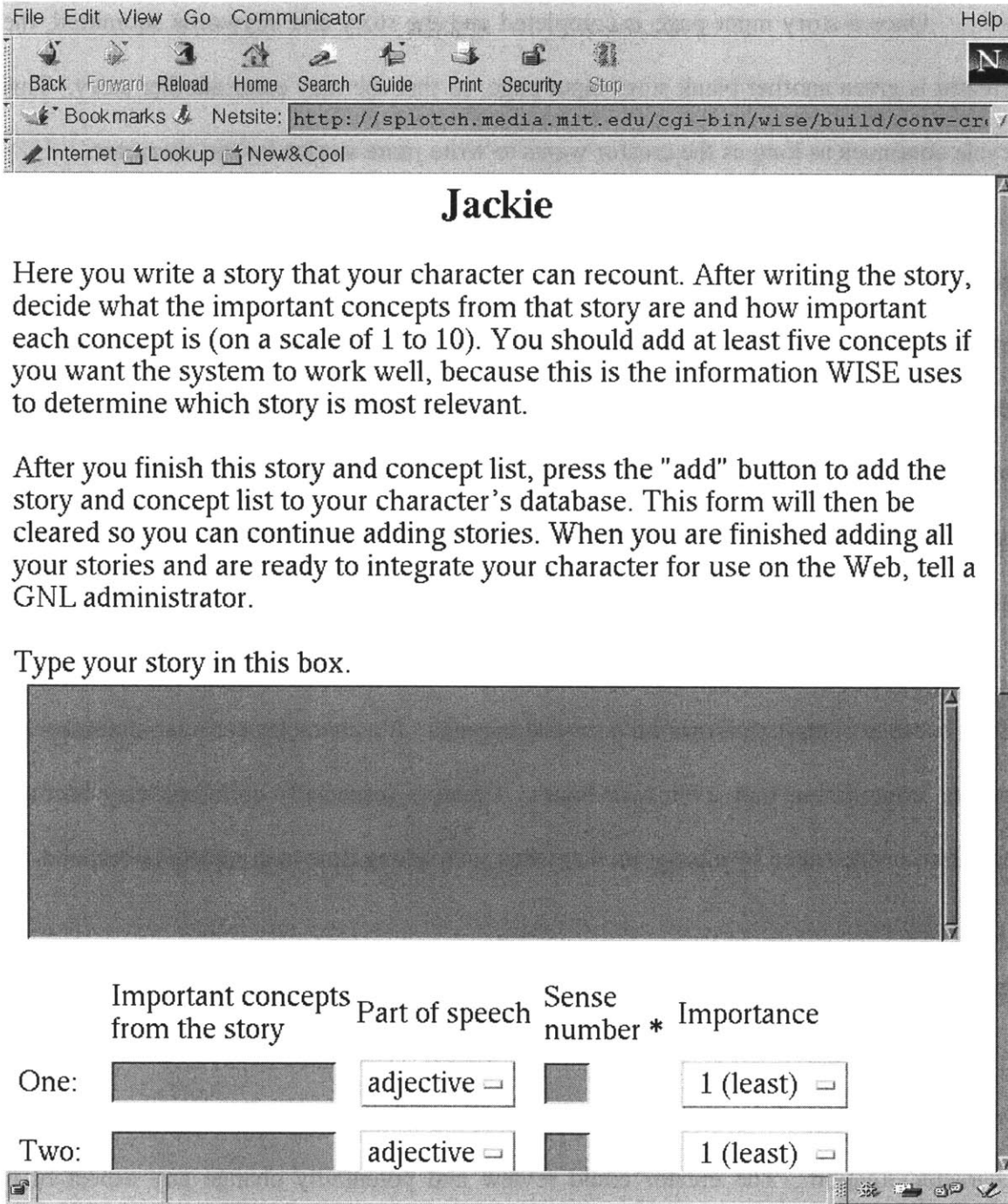


Figure 2-5: Character creation: Adding stories

Once a story input page is completed and the story and keywords submitted, the creator is given another blank story input page, so that she may enter another story. This cycle continues as long as the creator wants to write more stories for the character.

After all these steps have been completed to the creator's satisfaction, the character must be compiled. Compilation consists of the following stages: WordNet expansion, which adds the synonyms, hypernyms, and hyponyms of each keyword to the original keywords; verb extraction, which adds the verbs in each story to that story's list of expanded keywords; stopword culling, which eliminates very common words from the resulting list; and stemming, which strips suffixes from that list, so that the matching algorithm can match different forms of the same words.

Like preparation, compilation must be done by a WISE administrator. This is done because compilation may take several minutes. If a character has a large number of stories, compilation may even take hours. Creators (especially children) may become confused or frustrated by a program that takes such a long time to complete or respond.

2.3 Character Edibility

Editing characters is very similar to creating them. A creator can go to a certain web page to begin editing a character, and can then follow the same steps as she would in character creation. The creator could review and potentially change any aspect of a character's aesthetics or conversation, using pages identical to those of the creation process, but with the existing information already filled in. The creator then goes to story editing (see Figure 2-6), where she may review and edit any existing story and/or its keywords, or add a new story altogether.

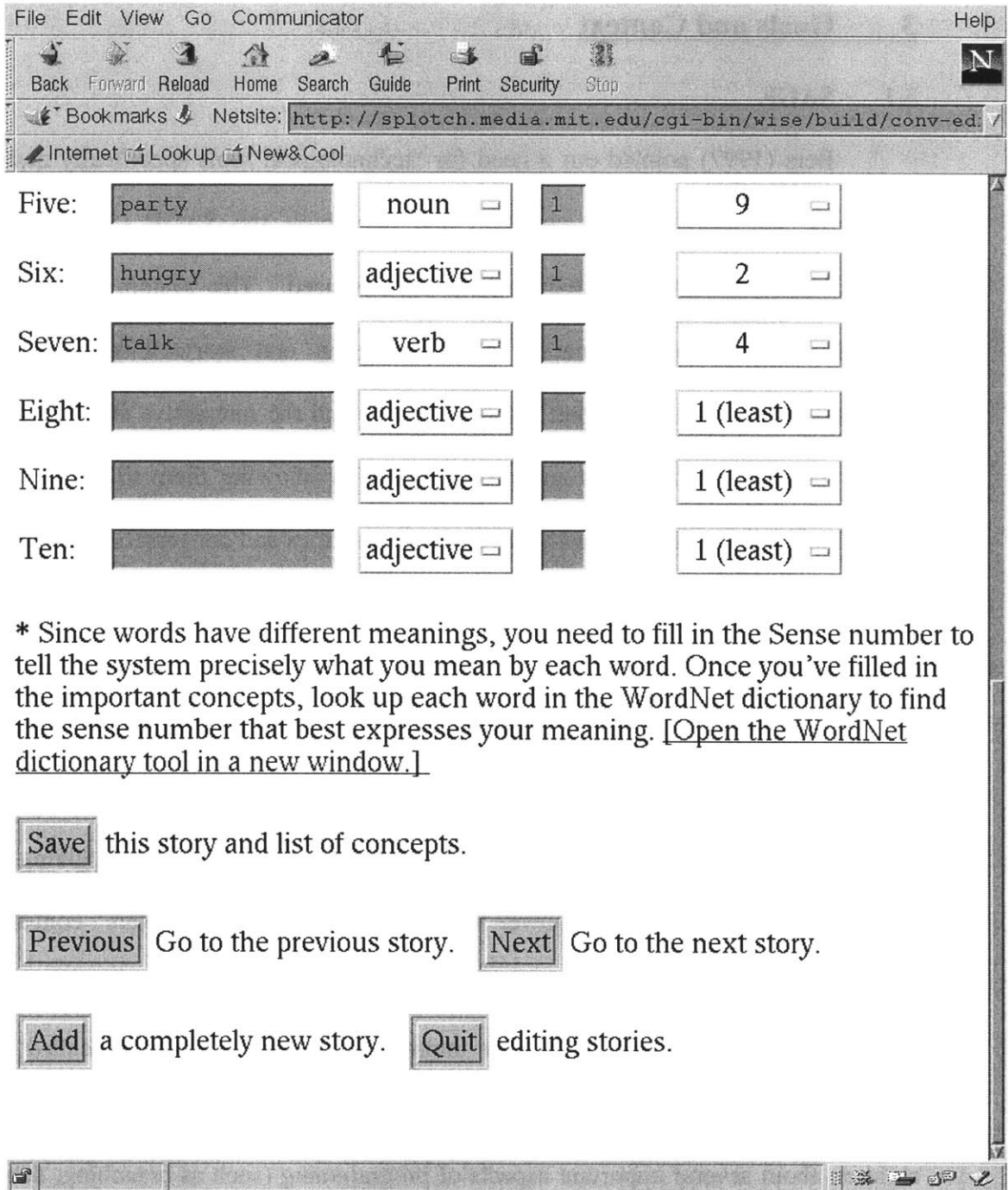


Figure 2-6: Character Editability: Story editing

3 Goals and Context

3.1 SAGE

Bers (1997) pointed out a need for “technological tools specifically designed to encourage self-reflection” in children, and then built the SAGE (Storyteller Agent Generation Environment) system to help fill this need. This system encourages self-reflection through a combination of conversation and storytelling (conversational personal storytelling) on the part of both the user and the interactive storyteller. It also teaches children about stories and programming by allowing them to create their own storytelling characters, along with that character’s stories and conversational structure.

The storytelling aspect of SAGE allows children to explore themselves and the world around them. In the process of telling personal stories to a character, as well as in writing stories for their own characters, children get a chance to reflect on and write about their experiences and the issues that concern them.

SAGE focuses heavily on conversation as well as on storytelling, giving creators a great deal of freedom and power in programming their characters’ conversational behavior. The fact that creators can interact with their own characters, make adjustments, and then renew interaction with their improved characters (a process called iterative design) is one of SAGE’s most powerful features. Bers found that iterative design taught children about several important aspects of programming (such as branching, debugging, knowledge representation, and, to some extent, abstraction) without their knowing it. It also helped them develop a concept of self that was separate from the concept of the user of the character.

Bers found that, in the process of decentering, iterative design in SAGE was reinforced when a person other than the creator interacted with a character. Children who created characters would often find communication bugs when they realized that other people would not necessarily give the same answers to the characters in conversation that they themselves would.

SAGE has limitations, however. Foremost among these is the cost of giving creators so much power and flexibility in building characters' conversational structure: the SAGE programming language, based loosely on LISP, is very non-intuitive and complex for character creators who have no programming experience. In addition, the characters created in the SAGE system are relatively inaccessible, in that each character is tied to the computer on which it was developed. This makes it difficult to share characters. Finally, the matching algorithm that chooses an output story based on the user's input is a simple one with little intelligence.

3.2 WISE: a successor to SAGE

Some of SAGE's limitations (namely programming complexity and inaccessibility) were addressed when a web-based storytelling system was envisioned. The WISE (Web Interactive Storyteller Environment) system, the first version of which was developed in Spring 1998 by Nick Montfort, Aalok Shah, and Kimiko Ryokai, is a simplification of SAGE that is designed to be more accessible and easier to use. Accessibility is achieved by making WISE a web-based system, requiring only that a user have a web browser. The fact that all characters are publicly available from one web server makes sharing characters trivial. The focus of WISE was shifted away from

conversation in order to create an case-of-use that SAGE does not have. The conversation between the character and the user is restricted to a linear interaction (i.e., WISE characters do not have the branching conversation trees that SAGE characters do). In WISE, the creator writes a script for each conversational turn from which the character never deviates.

Thus, WISE exists for many of the same reasons as SAGE (that is, to allow children to explore themselves and their world through conversational storytelling). In addition, WISE exists to provide a far more accessible and simpler interface for character creation, and to allow creators to effortlessly share their work with others around the world.

4 Literature Review

The WISE system and the modifications described in this thesis fall at the intersection of the following fields: cognitive development (specifically, in the area of children's ability to understand stories and resolve polysemy), natural language processing, and information retrieval.

4.1 WISE and the User

WISE is intended to address a number of areas of cognitive development in children:

- 1) semantic knowledge, especially word sense disambiguation, through the selection of story keywords;
- 2) the ability to find relevance in the stories that a WISE character returns;

3) story writing, understanding, and interpretation through the adding of stories and relevant keywords to a character's database.

Research suggests that addressing these areas is appropriate for children at the fourth or fifth grade level, and that these children may benefit from interaction with WISE.

Durkin (1986) notes the importance of vocabulary development during the school years, and points out that issues other than lexical addition need to be addressed. While some children have the ability to disambiguate homonyms in context by age 7 (Ceci and Howe, 1978), it may take much longer to do the same for polysemous words (Durkin et al, 1986). The current version of WISE uses the cycle of iterative design to provide an opportunity for children to disambiguate word senses when entering story keywords and then to test that disambiguation by interacting with their characters.

The listener or reader of a story relies on the storyteller to provide a context in which he/she may make sense of potentially ambiguous statements. Children understand that an utterance may have different interpretations by the age of 6 (Olson 1990), and, according to Robinson and Whittaker (1986), they are able to consciously resolve such ambiguity to some extent by the age of 7 or 8. The relationship between storyteller and listener or reader can be damaged, however. Ackerman (1981) points out that children who are led to mistrust a speaker were more aware of the ambiguity of the speaker's statements. Bers (1997) points out that her subjects exhibited a wide range of trust for the SAGE storytellers. That trust was occasionally negatively affected by responses which showed a lack of intelligence on the part of a SAGE character. Natural language processing would allow a character to make better use of text, and therefore project a more

consistent image of intelligence. Thus, adding text processing to WISE helps prevent the development of mistrust for a Wise character on the part of a user.

Children are also developing the ability to understand stories in primary school. Galda (1984) notes that children in second grade comprehend the motivations and emotional responses of story characters. According to Olson (1990), however, children are still learning in high school to understand the speech acts and mental states of characters in stories that test metalinguistic knowledge. Thus, children are developing the ability to pick out the most important aspects of a story. As in the case of sense disambiguation, the iterative cycle in the current version of WISE allows children to attempt to interpret a story and then to test that interpretation through character interaction.

4.2 The Technological Basis of WISE

WISE relies on tools and techniques from the fields of Natural Language Processing and Information Retrieval.

The two key tools from natural language processing that WISE uses are the WordNet lexical database and the Constraint Grammar Parser of English (ENGCG). The synonymy, hypernymy, and hyponymy relationships of WordNet are used to expand story keywords. Resnik (1995) and Klavans and Kan (1998) warn that word senses are difficult to disambiguate, and that without disambiguation, WordNet introduces a great deal of noise during the process of expansion. WISE avoids this problem by asking the creator of a character to indicate the intended sense for each story keyword. The Constraint Grammar Parser of English is used to tag the characters' stories with part-of-

speech information in order to extract the verbs. The ENGCG is accurate (only 2-2.5% incorrect tags) and extremely fast for a rule-based parser (it can analyze up to 400 words per second on some machines), though it does leave some ambiguity unresolved (Tapanainen and Jarvinen, 1994).

Manaris and Slator (1996) have called for “user-driven” interactive natural language processing systems with real-time or near real-time performance. WISE is intended to take a step in that direction by offering speed and ease-of-use. The most successful NLP systems to date, however, work in tightly restricted domains, and contain a great deal of knowledge about those domains (see Cavazza and Zweigenbaum, 1992, for an example of a system that understands certain medical texts).

WISE, as an interactive storytelling system, must work with a free text query (the user’s interaction) in an unrestricted domain to find relevant documents (in this case stories). In this light, it makes sense to examine some Information Retrieval techniques for use with WISE. Sussna (1993) argues that without semantic knowledge (represented in WISE by WordNet), recall and precision in an IR system cannot be improved at the same time. Though high recall is not a direct concern for WISE (only one story is returned to the user), it is felt that semantic knowledge can help WISE maintain a high precision by helping a character return a relevant story. Papka and Allan (1998) have shown that multiword queries (that is, queries treated as phrases rather than as separate search keywords) greatly enhance precision. Tombros and Sanderson (1998) have used a method of sentence extraction as a way to summarize a document, displaying exactly how the document relates to the user’s query. Summarization is not implemented in

WISE, however, because it is believed that the user should be free to make her own connections between her input and the character's story.

5 The Limitations of WISE

WISE, as it was first implemented, had a number of limitations. Foremost among these were the lack of iterative design, and the loss of some of the context that entices users of SAGE to suspend their skepticism of story-matching.

5.1 Loss of Iterative Design

As stated above, WISE was conceived as a simpler version of SAGE. To achieve this aim, it was decided that WISE should narrow its focus to one of the two fundamental aspects of SAGE: either the responsiveness of the character during conversation or the story listening and telling. It was felt that, while a character's responsiveness during conversation is an extremely valuable feature, the exchange of stories is the feature that draws users and, in fact, justifies the whole interaction. In addition, building conversational structure had proved to be the most complex and non-intuitive part of creating a SAGE character. As a result, WISE characters are not responsive to user input during conversation, and do not deviate from a linear conversational script written by the creator. This decision seems to obviate the need for SAGE's iterative design. In SAGE, iterative design is linked to the character's conversational structure. The decentering that Bers observed occurred as her subjects built and improved their character's conversational ability. In the first implementation of WISE, therefore, the ability to edit a character was dropped.

This lack of editability, while justified by the fact that iterative design has less of a place in a WISE character's conversation than it does in SAGE, is a serious strike against WISE's intended ease-of-use. Mistakes could only be fixed by recreating the character.

5.2 Loss of Context

Bers found that story matching in SAGE need not be very sophisticated (and could even be random). Users who interact with SAGE often make connections between their experience and the story returned by the character. This is because SAGE characters provide an "appropriate socio-cultural context and situation" (Bers 1997). That is, a user of SAGE understands that the character (often based on traditional sages who teach through parables and fables) will not return direct advice, but stories which the user must interpret for herself and apply to her own situation. The user is therefore willing to make connections even when the story has no obvious relevance. This suspension of skepticism occurs partly because of the mere existence of a character within the framework of interaction. From the user's perspective, there appears to be an intelligent being deciding on the stories being returned. This illusion of intelligence is enhanced by the character's responsiveness during conversation.

WISE provides the cultural context of a sage character who questions a user and then responds to her situation through a story. Due to the simplification of conversation, however, WISE characters cannot provide the specific feedback and responsiveness that SAGE uses to engage the user. WISE characters must therefore maintain their credibility by returning stories that are more clearly relevant to the user's situation.

The first version of WISE uses a story matching algorithm that depends on keywords specified by a character's creator when she writes stories for that character. Each story is associated with between five and ten words that the creator feels expresses the meaning of the story. The creator is also asked to give a part of speech, a word sense, and an importance for each keyword. These keywords are expanded using WordNet. SAGE's story matching algorithm is similar in that it uses WordNet to expand story words. It does not, however, ask the user for word senses, and therefore runs into trouble during expansion. WordNet returns synonyms, hypernyms, and hyponyms that are related to unintended and often obscure meanings of the words it expands. Thus, by asking the user to narrow its word expansion, WISE has a slightly more effective story matching algorithm. According to several adult users, however, the stories returned in WISE do not seem very relevant to the user's input. This indicates that the context provided by WISE characters (as opposed to that provided by SAGE characters) is not sufficient to get the user to connect a character's story to her input. Therefore, WISE needs improvements in its story matching algorithm.

6 Solutions

6.1 Regaining Iterative Design

Iterative design in SAGE is associated with improvement in conversational structure. The presence of editability in WISE would allow users to fix mistakes in the simple linear conversations they design, as well as in the keywords they associate with stories, and make their characters better storytellers.

While the conversational aspect of WISE is very limited in its function, it is still useful for evoking information and stories from users. As a character's stories are written and rewritten, the focus of the character might change, and the creator may wish to evoke different information from the user. The creator may also write conversational turns that do not evoke information, and may leave the user confused as to how to respond to the character. Bers observed this phenomenon in the creation of SAGE characters.

Editability would also help character creators choose appropriate keywords for their stories. When interacting with her own character, a creator may find that her character does not return the stories that she thinks are appropriate to given inputs. With editability, the creator would be able to go back and examine the keywords she associated with any of her stories, and change any aspect (word, part of speech, sense, or importance) of any keyword. She would then be able to retest her character, make more adjustments, and so forth, until she is satisfied with the way her character works. In the process of compilation, changes in keywords will affect story matching, by changing the large set of words returned in WordNet expansion, and/or the weights assigned to them. In the process, she would get a sense of how matches are made between stories, and would therefore have a better sense of how to improve her character on each cycle of iteration. Thus, the concept of iterative design would follow WISE's shift in focus from conversation to narrative. Rather than giving creators a great deal of intuition about programming, however, it would force them to think about the stories they write, the words they choose to represent those stories, and the relative importance of those words. Creators would also gain a greater understanding about the differences between word senses.

Editability brings with it the ability to correct errors, change the focus of a character, and iteratively make the character a better story matcher. Editability is therefore implemented in the current version of WISE.

6.2 Processing of Stories

I claim that the weakening of context and of the illusion of intelligence that differentiates the conversational aspect of WISE from that of SAGE needs to be compensated for by improvements in the story matching algorithm.

The approach I have taken to this issue in the current version of WISE consists of automatic verb extraction and keyword stemming. During the process of character compilation, the verbs from each story are extracted and added to the expanded list of keywords. The verbs are automatically given weights on the same scale as user-chosen keywords (from 1 to 10, with 10 being “very important”). Different versions of the compilation algorithm assigned verbs different weights: one version assigned low weights (about 0.25 on the above scale), one version assigned middle weights (5 out of 10), and the last version assigned high weights (10 of 10). All of those words, expanded keywords and extracted verbs alike, are later stemmed.

I chose to use verb extraction because it has successfully been used in information retrieval for the purpose of indexing news stories. I chose to stem keywords instead of inflecting them in the hope that it would reduce the size of match lists while allowing the matching algorithm to make connections between certain related words that are different parts of speech.

I hypothesized that stemming would allow the matching algorithm to make connections as well as or slightly better than inflection did, while keeping the match lists small and the compilation fast. I also hypothesized that the algorithm that assigns the middle weight to story verbs would return the stories most relevant to user input. I believed that the middle weight would allow the matching algorithm to match elements of plot captured in the verbs without overwhelming the important concepts expressed by the keywords.

6.2.1 Verb Extraction

In a number of ways, story matching is similar to information retrieval. Given user input, (a query in information retrieval) the matching algorithm must find the most relevant documents from a large database of documents. One technique used in IR is extracting the verbs from a query for use as keywords in a document search. Klavans and Kan (1998) used verb extraction to successfully classify newspaper articles according to genre.

Ideally, parables and fables should be matched according to moral rather than plot. In the case of interactions with SAGE and WISE, however, users often seek advice. During the course of interaction, users relate stories that describe their situation in the hope that a similar story will have a relevant moral. In these cases, the user's story, like a newspaper article, consists of a series of events, and these events may well be captured in that story's verbs.

The decision to extract verbs from each character story and add them to the list of keywords for that story was made as a result of these observations. There were some

concerns, however. Klavans and Kan point out that their attempt to classify articles using WordNet to expand their verbs failed because there was no sense disambiguation in their system, and because WordNet's tree of verb relations is extremely short and broad. These problems combined to give Klavans and Kan a large number of incorrect classifications based on matches on unintended meanings of certain verbs. This concern is addressed in WISE. Like the system built by Klavans and Kan, WISE has no way of automatically disambiguating word senses. However, WISE does not expand these extracted verbs through WordNet. This is acceptable, since the matching algorithm does not depend only on verbs, but also on the keywords selected by the user (which are expanded using WordNet, but avoid the disambiguation problem because their senses are specified).

6.2.2 Stemming

Each of the expanded keywords and the verbs extracted from stories are stemmed heuristically using the Porter Stemming Algorithm. This is done so that different inflections of the same word (say "deceive" and "deceiving") are matched by the story matching algorithm. In the first version of WISE, the same effect was achieved by heuristically inflecting all keywords. Stemming was decided upon in the next version to control the size of the match lists (since all the inflections of one word take up more space than a single stem that corresponds to all the same inflections) and therefore the time of compilation. It was also hoped that a number of matches that could not be made through inflection (such as between the noun "magnet" and the verb "magnetize") could be made through stemming.

Ultimately, however, stemming failed to improve the matching algorithm for reasons explained later (see Section 9.1). In the last phase of experimentation, WISE returned to inflection.

7 Technical Design and Implementation of WISE

7.1 Design Decisions

One of the goals in building WISE was to create a version of SAGE that was highly accessible. Therefore, WISE was designed so that interaction with and creation of characters in the system would be similar to the corresponding actions in SAGE. This was largely accomplished in the design of the interaction phase, but there were many exceptions made in the design of the character creation phase. These exceptions reflected differences in the data representation of characters and the fact that WISE would be a web-based system, as well as the desires to focus on storytelling rather than conversation and to simplify the process of character creation.

A character, as referred to in this document, is a set of HTML pages and a Perl script that manages the interaction phase (and includes the character's story matching algorithm).

The interaction phase of WISE differs in one major way from that of SAGE. SAGE used an interactive stuffed animal with voice synthesis (what Bers (1997) calls a "soft interface"). SAGE's soft interfaces were very specialized, and therefore were not incorporated into WISE in the interest of accessibility.

The creation of characters in WISE was designed to be much simpler than in SAGE. Character creators, who may be accessing WISE from anywhere, and who may

not have any guidance in using the system, must be able to create characters very easily. Thus, the creation pages follow each other in a logical order. In keeping both with simplicity and with WISE's focus on storytelling, the conversational element of WISE is far less complex than that of SAGE. SAGE's programming interface required that children learn a Lisp-like language for pattern-matching and template-filling, so that the SAGE character could dynamically and appropriately respond to a user during interaction. WISE characters are thus less responsive during conversation, but are far more intuitive to create. WISE does have one disadvantage that SAGE does not: looking up word senses may be tedious for creators. However, the creation process is still fairly intuitive and easy for creators to learn. The benefits of asking for the part of speech and word sense extend to both the creator, who must think about the meanings of her stories and then must learn to resolve polysemy, and to the WISE system, which is given a great deal of information that cannot be determined automatically, given the current state of natural language processing.

In keeping with the goal of WISE to emulate SAGE, creators have been given the ability to easily edit their characters. This ability was implemented in WISE recently. Bers (1997) found that the ability to edit characters was extremely useful, and helped children who used SAGE learn the concept of iterative design.

7.2 Implementation of WISE

7.2.1 Tools Used in WISE

The core of WISE is not very large or complex. However, it offers character users and creators a great deal of flexibility and speed. It also allows creators to specify

the exact meanings of their character's stories using keywords and their senses. During compilation, WISE can intelligently and automatically look for a number of features in characters' stories. In order to do all this, WISE draws on a number of existing tools and technologies.

7.2.1.1 HTML

Most of the features in WISE are implemented using HTML and Perl. HTML, used for virtually all of the Internet's web pages, consists of a large set of tags that can be used to manipulate the way a document is displayed on a web browser. Thus, pages written in HTML serve as the graphical user interface (GUI) for WISE. Most HTML tags (and all tags used in WISE) are uniform across web browsers, making WISE browser-independent. Writing and manipulating HTML can be done very easily, using only text. Thus, WISE can easily and accurately report the state of character creation or interaction by dynamically generating pages based on HTML templates. It can also gather information from the user or creator through the use of HTML forms that can be submitted to an active program for processing.

7.2.1.2 Perl

The programs used to manipulate the HTML in WISE, as well as store data, control interaction, and manage other tools, were all written in Perl. Perl is a language that is commonly used in web applications. Despite the fact that it is an interpreted language, it is extremely fast and easy to use for text processing. Since WISE focuses on conversation and stories, and uses text-based HTML tags, the ability of the developers of

WISE to easily write code to process text is extremely important. In addition, many potential users of WISE may have extremely slow Internet connections. Data transfer both to and from those users is fairly time-consuming. In order to keep users from losing interest, a web-based system must minimize the time it adds to data transfer time. It is therefore also important for WISE to be able to read input, process it, and generate output as quickly as possible. As an interpreted language, Perl is also written in text, and is therefore easily manipulated. That is, one Perl script can be used to modify the text of another one. This feature is used in character creation. In this phase, the same Perl script that writes a character's HTML pages also uses a template to generate the Perl script that will run that character during the interaction phase.

7.2.1.3 WordNet

The tool that allows creators to specify senses for their keywords is WordNet. This tool, developed by George Miller et al (1993), is a lexical database that relates words through synonymy, hypernymy, and hyponymy. WordNet has been criticized for introducing a great deal of noise into searches when used in query expansion in information retrieval (Klavans and Kan, 1998). This is because WordNet has an exhaustive set of senses (including many obscure ones) and relations to other words for each word in its database, and because there is no good method for automatic sense disambiguation. When, as in WISE, the sense and part of speech of a word are given, WordNet becomes an extremely valuable tool. It allows a creator's story keywords to be accurately expanded into its synonyms, hypernyms, and hyponyms. This expansion is the first step towards creating a match list for the matching algorithm to use. It increases

the probability that any given keyword will be matched by a user story word with a similar meaning.

7.2.1.4 The Constraint Grammar Parser of English

The Constraint Grammar Parser of English, or ENGCG, is an accurate parser of English sentences (Tapanainen and Jarvinen 1994). Since it was felt that creators' keywords alone were not producing good story matches, the information retrieval technique of using verbs was added to the system. The 'Lite' version (part-of-speech tagging only) of ENGCG was incorporated into the compilation phase of WISE in order to extract and tag verbs from characters' stories. ENGCG was chosen because of its speed and accuracy. The Lite version was chosen because ENGCG's only task in this implementation was to extract verbs from characters' stories. ENGCG also has the ability to stem words. Because of a licensing agreement, however, the version of ENGCG we used could not be run through a web interface. Thus, it could not be used to analyze the user's input in the interaction phase. Stemming was therefore done by the Porter stemming algorithm.

7.2.1.5 The Porter Stemmer

The Porter algorithm is a heuristic stemmer that uses several general rules to strip suffixes from words. This algorithm is not always accurate, but it is very fast and requires no large lexical database. It was used to stem the words in each character's match list and the words in the user's stories. This was done so that user's story words could be matched against characters' keywords regardless of inflection.

7.2.2 Compilation

The compilation of a character creates a match list out of keywords and story texts using WordNet, the ENGCG-lite tagger, and the Porter stemmer. This procedure is accomplished in several steps: WordNet expansion, verb extraction, the exclusion of common words (stopwords), stemming, and finally, the consolidation of each story's keywords into one easily searchable file. These steps, described below, are illustrated by actual results in Appendix C.

7.2.2.1 WordNet Expansion

During the character creation phase, each story that the creator writes for a character is saved in its own text file. The keywords for each story are also stored separately.

In this stage of compilation, WordNet is used to look up the synonyms, hypernyms, and hyponyms of each keyword that is either a noun or a verb. This is done so that the matching algorithm can link a story keyword to a related word typed during interaction by the user (for example, between “anger” and “fury”) without the user having to type the original keyword exactly. WordNet includes hypernym and hyponym relationships only between nouns and verbs, so for adjective keywords, only synonymous words are looked up. The sense of each keyword given by the creator is also specified to WordNet using a command-line option. This ensures that WordNet's expansion focuses solely on the intended meaning of each keyword and returns only synonyms, hypernyms, and hyponyms related to the original keyword by the same sense. Words related to the

original keywords by unintended meanings are excluded, eliminating one of the problems that plagued SAGE's story matching algorithm.

The weighting system is changed slightly in this stage to allow for greater granularity. Hypernyms and hyponyms can help capture a more general or more specific aspect of a keyword. However, while a word's synonyms by definition mean the same thing as that word, hypernyms and hyponyms do not. In addition, as they get farther from the original word (e.g. the hypernym of a hypernym) they become too general or too specific to be of much use. Therefore, these words are given a lower weight than the original keyword and its synonyms. Near hypernyms and hyponyms (i.e. within one degree of the keyword) are given half the weight of the keywords, while far hypernyms and hyponyms are given a quarter of the weight. (What actually happens is that keywords and their synonyms are given four times their original weight, while near hyper- and hyponyms are given double the weight, and far hyper- and hyponyms are given the original weight. Again, this is done to increase granularity in the interest of precision.)

In this stage of compilation, word repeats are resolved by summing the weights of all copies of a repeated word. This is done so that concepts that appear frequently in the expansion of user-selected keywords are given more importance.

The results of this stage of compilation are stored separately from the keywords, with one file of expanded keywords corresponding to one character story.

7.2.2.2 Verb Extraction

In the next stage of compilation, ENGCG-Lite is used to extract all the verbs from a character's stories.

Each story is processed by ENGCG-Lite, producing a list of each word in the story and its part of speech. Those words that are classified as verbs are taken and inserted into the expanded list of keywords with a certain weight. The weight given to automatically extracted verbs is adjustable. The story-matching algorithm was tested using three different weights for verbs: the highest possible value (corresponding to a ten in the keyword selection stage of character creation), a middle value (corresponding to a five), and the lowest possible value (corresponding to 0.25). See Chapter 8 for results comparing the different weightings. The automatically extracted verbs are not expanded using WordNet, because with no sense information, expansion would introduce a large number of words linked by unintended meanings.

Repeated words are again consolidated. In this stage, however, the resulting weight is the higher of the two values instead of their sum. This is done so the effects of automatic verb extraction are clear. This metric gives priority to the user's choice when the verb extraction is given a small role in story matching (i.e. a low weight), but gives priority to the automatic weighting when verb extraction is to have a larger role. This also avoids the possibility that story matching will be dominated by one or two verbs that are repeated several times in the same story.

7.2.2.3 Culling Stopwords

A number of words that are added to the keyword list in the expansion or extraction stages are extremely common and are not necessarily relevant to the meaning of the story they come from. A good example is the verb “to be.” These words, if left among the keywords, would cause a large number of false matches. Therefore, they need to be excluded.

In Aesop, for example, 64 original keywords out of 975 (6.6%) are stopwords. During the process of expansion and verb extraction, a large number of stopwords are introduced, raising the proportion to 1612 stopwords out of 12927 total words (12.5%).

Culling is done using a large list of about 1000 common English words, including inflected forms and related words. Each of the expanded keywords (including the verbs extracted from the stories and the words added during WordNet expansion) is compared against this list, and is removed from a story’s expanded keyword list if there is a match.

7.2.2.4 Stemming

In the next stage of compilation, the Porter algorithm is used in conjunction with WordNet’s inflection lists to stem the expanded keywords.

WordNet includes several word lists that associate inflected words with their uninflected forms. These lists exist for several parts of speech. In this stage of compilation, each keyword is looked up in the appropriate inflection list, and its deinflected form (if found) is noted. If a match is not found, then the keyword is sent through the Porter stemmer. The reason that both stemming options are used is because

the WordNet lists focus more on irregular words, while the Porter stemmer (as a heuristic algorithm) is made to handle regular ones.

As in verb extraction, the repeats that occur are resolved by keeping the word with the higher weight, and not the sum, as in WordNet expansion. The results of this stage are stored in separate files, with each file corresponding to one of the character's stories.

7.2.2.5 Keyword Sorting and Collection

In the final stage of compilation, a large match list is compiled from the final lists of expanded, extracted, culled, and stemmed words. The keywords, up to this point, are stored in separate files, corresponding to the story they came from. This stage exists so that the story matching algorithm that runs during the interaction phase can match keywords easily and quickly.

The keywords in every story's list are all stored into a single file. Each keyword is associated with the story it came from and its weight. This file is then sorted alphabetically. Repeats are no longer resolved in the same way as in previous stages, since at this stage, each occurrence of a word comes from a different story. Instead, the repeated keyword is stored only once in the match list, and is followed by the list of all the stories it is associated with and the weights it has in each. For example, if the word "admit" is associated with stories 21, 24, and 64, from which it has weights of 7, 9, and 3, respectively, then its entry in the match list looks like the following:

admit,21=7,24=9,64=3

This format allows the matching algorithm to work more efficiently.

7.2.3 Interaction

The interaction phase is designed to be as fast as possible in order to keep users' interest. Most of the interaction phase consists of collecting user input from text boxes and information from hidden HTML fields. This is done using HTML forms, which invoke programs (in this instance Perl scripts) that use the information collected as input. During the conversation, the user's input is simply recorded and accumulated (with input from previous turns), and the next page of the conversation is displayed. After the last conversational turn, the user's input is processed and given to the story matching algorithm, which selects an appropriate story from the character's database. The story is then displayed to the user. After the interaction, the user has the option of returning to the first turn for a new interaction with the same character.

7.2.3.1 The Story Matching Algorithm

As the user interacts with a character, her inputs are accumulated. After the last interactive turn, those inputs are fed to the matching algorithm.

First, the matching algorithm stems each word of the input using the Porter algorithm. Next, an array of relevance scores is initialized. Each slot in this array corresponds to one of the characters' stories, and is initially set to zero.

Next, each word in the user's input is looked up in the character's match list. A binary search algorithm is used for this lookup, so that each lookup takes a minimal amount of time. When a match is found, the stories that the matched word is associated have their relevance scores increased by the appropriate weight. So for example, if the

matching algorithm searched for the word “admit,” it would find the entry described above:

admit,21=7,24=9,64=3

The relevance score for story 21 is then increased by 7, that of story 24 is increased by 9, and the score for story 64 is increased by 3.

It would be possible to remove stopwords from the user’s input in order to reduce the number of accesses to the match list. However, searching for a word in the list of stopwords is no different than searching for it in the match list. The matching algorithm would in fact have to search for every input word in the stopword list before searching the match list. Removing stopwords would therefore make the story matching algorithm slower.

After all of the input words have been searched for in the match list, the algorithm determines which story has the highest score. In case of a tie, the story indexed with the lower number (story number 24, as opposed to 64) is returned to the user. The story with the highest score is then inserted into the character’s last conversational turn.

8 Experimentation and Results

Experimentation for this thesis consisted of testing the modifications to WISE’s story matching algorithm. This was done through surveys completed by college students. Two surveys were done. The first tested both changes to the matching algorithm (verb extraction and stemming), while the second tested verb extraction alone.

8.1 Methodology

The story matching algorithm was tested to determine whether adults (in this case college students) saw more relevance in stories returned by the modified matching algorithm than in those returned by the unmodified one. The matching algorithm was tested using a WISE character named Aesop, whose story database consists of 82 fables (commonly attributed to the character's perhaps legendary namesake). The testing was also intended to determine the optimal importance to be given to automatically extracted verbs. Each survey was therefore made up of five questions, each in the following format: one story written and submitted by a college student, followed by two, three or four fables (please see the appendices). For each original student story, the person being surveyed was asked to rank the fables that followed in order of their relevance to the original story (with 1 being most relevant). Ties were allowed. They were also asked to provide a brief explanation of their reasons for choosing that ranking.

The original stories were submitted by college students who had no knowledge about WISE or about the algorithm being tested. Each story was given as input to four different versions of Aesop's story matching algorithm. The first version was the unmodified algorithm, based on keywords and inflection. The other three were modified versions of the algorithm (with stemming for the first survey and inflection for the second). Out of those three, one version (the low weight version) gave automatically extracted verbs an importance equivalent to 0.25 (on a scale to ten), the second (middle weight) gave an importance equivalent to 5, and the third (high weight) gave an importance equivalent to 10. All the fables returned by the different versions of the algorithm for the same original story were used in the survey. In the common case where

the same fable was returned by more than one version, the corresponding survey question contained fewer fable options. Stories that evoked the same fable from all versions of the algorithm were not used.

8.2 First Survey and Results

The first survey tested the effect of both stemming and verb extraction on story matching. The modified algorithms came up with matches based on a relatively small number of keywords with high weights, compared to the unmodified algorithm. (For a listing of words matched for one input story by the different algorithms, see Appendix D. Also, for an example of the keywords associated by myself and Nick Montfort with one of Aesop's fables, please see Appendix C.) In fact, many of the words matched by the unmodified algorithm were not matched by the modified ones. Of the original stories that were run through Aesop's matching algorithm, five whose returned fables seemed to have the greatest variety were used for the survey. Of those five stories, three were associated each with three fables, and the other two stories were followed by two fables each (see Table 8-1). Six M.I.T. undergraduates unfamiliar with WISE completed and returned the surveys.

The raw responses from the survey are shown in Table 8-2. From that data, rankings were determined for the algorithms themselves. Separate rankings were obtained for each original story. The algorithms were ranked in the following manner. Each time a fable received a first place ranking (from the surveys) four points were given to the algorithm(s) that returned that fable. A second place ranking was worth three points, and so forth. Algorithms that returned the same fable for a given original story

Original Story	Tennis Tournament	Cooking Chicken	Transferring to MIT	Snorkeling	The Gate
Unmodified algorithm	The Labourer and the Nightingale	The Cat-Maiden	The Hare with Many Friends	The Fox and the Mask	The Bat, the Birds, and the Beasts
Low weight algorithm	The Shepherd's Boy	The Hart in the Ox-Stall	The Hare with Many Friends	The Crow and the Pitcher	The Fisher
Middle weight algorithm	The Frog and the Ox	The Frog and the Ox	The Hare with Many Friends	The Frog and the Ox	The Fisher
High weight algorithm	The Frog and the Ox	The Frog and the Ox	The Frog and the Ox	The Frog and the Ox	The Fisher

Table 8-1: First Survey: Original stories and associated fables

Original Story	Algorithm	Ranked First	Ranked Second	Ranked Third
Tennis Tournament	Unmodified	4	1	1
	Low weight	0	3	3
	Middle, high weight	2	3	1
Cooking Chicken	Unmodified	2	3	1
	Low weight	4	2	0
	Middle, high weight	0	1	5
Transferring to MIT	Unmodified, low, middle weight	4	2	
	High weight	2	4	
Snorkeling	Unmodified	4	2	0
	Low weight	2	4	0
	Middle, high weight	0	0	6
The Gate	Unmodified	1	5	
	Low, middle, high weight	5	1	

Table 8-2: First Survey: Fable rankings

Algorithm	No. of stories for which the algorithm's fable placed 1 st	No. of stories for which the algorithm's fable placed 2 nd	No. of stories for which the algorithm's fable placed 3 rd
Unmodified	3	2	0
Low weight	3	1	1
Middle weight	2	1	2
High weight	1	2	2

Table 8-3: First Survey: Algorithm rankings

given the same scores. Thus, for the story "Transferring to MIT," for example, the low, middle, and high weight algorithms each received

$$(4 \text{ points} \times 4 \text{ first place rankings}) + (3 \text{ points} \times 2 \text{ second rankings}) = 22 \text{ points}$$

while the unmodified algorithm received

$$(4 \text{ points} \times 2 \text{ first place rankings}) + (3 \text{ points} \times 4 \text{ second rankings}) = 20 \text{ points.}$$

Thus, for the story “Transferring to MIT,” the three modified algorithms tie for first place, while the unmodified algorithm comes in second. The number of such first, second, and third place rankings each algorithm received on this survey is shown in Table 8-3.

According to these algorithm rankings, the unmodified algorithm returned the most relevant stories (with three first place rankings and two seconds) followed very closely by the low weight algorithm (with three first place rankings, one second, and one third).

The reasons that respondents gave for their rankings are discussed further in Section 9.2.

8.3 Second Survey and Results

Heuristic stemming was found to have a number of faults, and the results obtained in the first survey were attributed to fundamental problems with that approach. (See Section 9.1 for a discussion of problems with stemming). Therefore, the second survey was designed to test verb extraction as the only modification to the matching algorithm. Stemming was replaced by the unmodified algorithm’s strategy of inflection. Again, the modified algorithms in general returned fables based on fewer matched keywords than the original. This time, however, that difference was not as pronounced. Three of the original stories used in the first survey were also used in the second. “Snorkeling” and “Tennis Tournament” were discarded because of a lack of variety among the fables

returned. In fact, all four algorithms returned the same fable (“The Fox and the Mask”) for “Snorkeling”. (In the case of “Snorkeling,” all four algorithms with inflection picked up on the word “deceiving,” while those with stemming did not. This example illustrates a deficiency in the Porter stemmer, discussed in Chapter 9.) These stories were replaced by two others that had more variety in the returned fables. In this survey, two of the original stories were associated with three fables each, two of the stories with two fables, and one story with four fables (see Table 8-4). Thirteen M.I.T. students (ten undergraduates and three graduate students) unfamiliar with WISE returned the surveys. Three respondents did not answer all the questions on the survey. Two of them left the fourth question unfinished, while the other did not complete the fifth question. The raw results are shown in Table 8-5.

Again, the algorithms themselves were ranked according to the fable rankings on the surveys. The same ranking system was used as on the first survey (see Section 8.2), and the results are shown in Table 8-6. The fables returned by the middle weight algorithm (with three first place rankings, one second, and one third) appear to be favored by the respondents, followed closely by those returned by the low weight algorithm (with three first place rankings, one third, and one fourth). The fables returned by the unmodified algorithm were clearly least favored.

Thus, the fables returned by algorithms that used verb extraction with low or middle weights coupled with keyword and verb inflection were preferred over the unmodified algorithm by the respondents in the second survey. This seems in part to confirm one of my hypotheses (as stated in Section 6.2): that the middle weight algorithm would return the most relevant stories. This plus the fact that respondents preferred the

Original Story	Dance Performance	Cooking Chicken	Transferring to MIT	The Departure	The Gate
Unmodified algorithm	The Bat, the Birds, and the Beasts	The Cat-Maiden	The Hare with Many Friends	Hercules and the Waggoner	The Bat, the Birds, and the Beasts
Low weight algorithm	The Bat, the Birds, and the Beasts	The Old Woman and the Wine Jar	The Lion and the Mouse	The Woodman and the Serpent	The Fisher
Middle weight algorithm	The Hare with Many Friends	The Old Woman and the Wine Jar	The Hare with Many Friends	The Hare with Many Friends	The Fisher
High weight algorithm	The Frogs Desiring a King	The One-eyed Doe	The Hare with Many Friends	The Wolf and the Crane	The Fisher

Table 8-4: Second Survey: Original stories and associated fables

Original Story	Fable	Ranked First	Ranked Second	Ranked Third	Ranked Fourth
Dance Performance	Unmodified, low weight	2	3	7	
	Middle weight	6	6	1	
	High weight	5	4	4	
Cooking Chicken	Unmodified	4	5	4	
	Low, middle weight	2	3	8	
	High weight	7	5	1	
Transferring to MIT	Unmodified, middle, high weight	3	10		
	Low weight	10	3		
The Departure	Unmodified	4	3	2	2
	Low weight	2	0	3	6
	Middle weight	4	3	3	1
	High weight	1	4	4	2
The Gate	Unmodified	1	11		
	Low, middle, high weight	11	1		

Table 8-5: Second Survey: Fable rankings

Algorithm	No. of stories for which the algorithm's fable placed 1 st	No. of stories for which the algorithm's fable placed 2 nd	No. of stories for which the algorithm's fable placed 3 rd	No. of stories for which the algorithm's fable placed 4 th
Unmodified	0	4	1	0
Low weight	3	0	1	1
Middle weight	3	1	1	0
High weight	2	2	1	0

Table 8-6: Second Survey: Algorithm rankings

inflecting unmodified algorithm over stemming algorithms seems to disprove my hypothesis that stemming would allow the matching algorithm to make connections between words at least as well as inflection.

As a result, WISE compilation in the future will include verb extraction, with verbs assigned the middle weight, and the inflection of all keywords and verbs.

9 Evaluation

Information gathered during the story matching process and through the two surveys justifies the results of the surveys; that is, that stemming is harmful to a keyword matching algorithm because of its inaccuracies, while verb extraction is somewhat helpful because users consider plot elements as well as morals when looking for relevance between stories.

9.1 Heuristic Stemming

Stemming was implemented in WISE for a number of reasons. Foremost among these was the hope that it would be at least as effective at matching keywords as inflection, by bringing different forms of the same word together to be matched on their stems. In addition, it was hoped that related words (nouns and verbs, for example, that represent nearly identical concepts) would be matched with each other through stemming. For example, the noun “magnet” and the verb “magnetize” are related. Inflection would turn “magnet” into “magnets” and “magnetize” into “magnetizing,” “magnetizes,” “magnetized” and so forth, in which case the verb (appearing in a story, say) would never match the verb (as a keyword, for example). Stemming, on the other hand, would reduce

“magnetize” to “magnet,” thus establishing the connection between noun and verb. Finally, stemming was implemented in order to use less space and computation time during compilation and story matching, since words could be stored as a single stem rather than a number of inflected forms that need to be searched through.

It was found, however, that heuristic stemming does not fulfill these hopes very effectively. One major reason for this is that stemming is just as incapable of correctly dealing with irregular words as heuristic inflection is. The verb “to sell” is an example. A heuristic inflector comes up with “selled” as a form of “sell.” On the other hand, the stemmer can do nothing to the word “sold” and therefore, in neither case can any connection between the words “sell” and “sold” be made.

In addition, several words that have the same roots also have vastly different meanings, and do not need to be connected in a matching algorithm. One such example from Aesop’s match list consists of the words “general,” “generate,” and “generous.” The Porter stemmer reduces all three words to the stem “gener.” While there may be some etymological link, the three words are semantically totally different. An inflector would leave the distinction between the three words intact.

Finally, the Porter stemming algorithm has a number of practical problems. While it is state-of-the-art as far as heuristic stemming, it still has a number of inconsistencies. Heuristic stemming is known to have difficulties with irregular words, but the Porter algorithm is occasionally inconsistent about even regular words. For example, the word “deceive” becomes “dece.” At the same time, an inflection of the same verb, such as “deceiving” becomes “deceiv.” In fact, this particular pair of stemmings kept the modified algorithms from matching a key concept (that “looks could

be deceiving”) in the original story “Snorkeling.” In this example, four words (“deceive,” “deceived,” “deceives,” and “deceiving”) are affected by the Porter stemmer’s inaccuracy. In Aesop’s match list, which contains 15,246 words, 2103 of them (or 13.8%) were affected by stemming inaccuracy in this way. This number does not include irregular words (such as “caught,” which should be matched with “catch”) or similar words that are different parts of speech (such as the verb “abandon” and the noun “abandonment”). Thus, a very significant percentage of the words in the match list are affected by stemming inaccuracy.

9.2 Verb Extraction

Bers (1997) intended that users who interacted with SAGE characters would be able to share their experiences and then receive a comforting story in return. She expected (and experimentally found) that children using the system would seek a deep meaning in the returned stories, and would often come up with advice or a connection to their own situations. WISE was also intended for that type of interaction, where the user does not intend to swap similar fables, but instead desires to interact with an intelligent and comforting storyteller who can listen to her experiences and respond appropriately. Verb extraction, therefore, was put into WISE based on the belief that user’s stories would be situations in search of a moral. That is, users interacting with a WISE character would expect a fable with a plot somewhat similar to their own experiences, and would apply to themselves the moral of that fable. Users would thus pay more attention to plot elements than to morals when deciding on the relevance of the story. The plot would be

the setup, showing the user that the fable really is relevant to her, while the moral would serve as an insight or advice for her.

The respondents who completed the two surveys had no context (in the form of an intelligent character offering advice) whatsoever that might lead them to see the original stories as requests for help. However, they still often looked at plot rather than moral to determine whether and why any particular fable was relevant to the original story.

The keywords (chosen by Nick Montfort and myself) for the majority of the fables in Aesop's database focus on the tales' morals without taking into account any elements of the plot. "The Frogs Desiring a King" is one such example, in which the Frogs repeatedly petition Jove to send them a king until Jove, out of annoyance, gives them a bird that eats frogs. The majority of that fable's keywords relate to government (such as "rule", "anarchy", and "government"). However, none of them endeavor to represent the persistence and determination of the frogs in getting a king from Jove. Similarly, the most important keywords for "The One-eyed Doe," which concerns a deer with one eye trying to escape her inevitable death, are related to destiny. Plot elements, such as the fact that the Doe lost an eye, are not addressed in the character's keywords.

Respondents to the surveys did not always stick as tightly to the moral of each fable. Thus, two respondents picked up on the determination of the frogs to have a king, and related that to the dancer's determination to put up a good performance. One recalled that both the frogs and the dancer had two chances to get what they wanted, and despite the fact that they got opposite results, this plot element was the connection between the original story and the fable. Another respondent noted that the plot of "The One-eyed Doe" could in some way have been a continuation of that of "Cooking Chicken," had the

cook not been as well trained. This respondent used that idea to justify his rating of “The One-eyed Doe” as the most relevant fable for that original story. Four respondents found relevance between “Cooking Chicken” and “The Old Woman and the Wine-Jar” in the protagonist’s actions of rushing to the sink and to the wine-jar, respectively.

10 Conclusion

SAGE was built to address a need for “technological tools that encourage self-reflection” in children (Bers 1997). A lack of accessibility and the ability to share characters with others, in addition to the difficulty of programming a character’s conversational interface were major drawbacks to SAGE. WISE was therefore implemented as a web-based successor to SAGE, focusing on storytelling as opposed to conversation. WISE, too had limitations. Foremost among those were the loss of the iterative design prized in SAGE and the weakening of the context in which storytelling characters operate.

These two deficiencies in the WISE system have been addressed in this thesis. Character editability has been given to WISE to restore SAGE’s cycle of iterative design, with its focus shifted from conversation to storytelling. At the same time, story matching has been addressed through the use of automatic verb extraction and weighting. Heuristic stemming was used in the first attempt to improve story matching, but that proved to be counterproductive.

The improvements to the story matching algorithm were tested using two surveys, each completed by several college students. Verb extraction (with the verbs automatically given a mid-range weight) as a supplement to keyword expansion was

found to be the most effective improvement to the story matching algorithm. It was therefore decided that the middle weight algorithm with inflection would be the default in the compilation of WISE characters. The results found in the surveys were then justified based on information gathered during the story matching process and from the respondent's explanations of their rankings.

11 Future Work

More work needs to be done in terms of user-testing WISE. A study should be done to determine the value of character editability in WISE. Observations made in workshops with adults indicate that editability will certainly be valuable in helping to fix mistakes, and probably also in getting character creators to become more effective at building and improving their characters.

WISE can also be improved in the area of ease-of-use. Several character creators have called for a simpler, less tedious method of looking up word senses and inputting keywords and their plethora of associated information.

The story matching algorithm could be improved further using semantic representations of both user and characters stories. That is, a semantic understanding of stories would better enable a matching algorithm to make connections between actions and outcomes of character and user stories.

Finally, improvements in story matching may not be the only way to make characters more believable. It may be possible and appropriate to strengthen the weakened context of interaction by giving each character a larger number of interaction turns.

12 References

- Ackerman, B. P. 1981. "Performative bias in children's interpretations of ambiguous referential communications." *Child Development* 52. pp. 1224-30.
- Bers, Marina Umaschi. 1997. *Soft Interfaces for Interactive Storytelling: Learning about Identity and Communication*. M. S. Thesis, Massachusetts Institute of Technology.
- Cavazza, Marc, and Pierre Zweigenbaum. 1992. "Extracting Implicit Information From Free Text Technical Reports." *Information Processing and Management* 28. pp. 609-618.
- Ceci, H. J. and M. J. A. Howe. 1978. "Semantic knowledge as a determinant of developmental differences in recall." *Journal of Experimental Child Psychology* 26. pp. 230-45.
- Durkin, Kevin. 1986. *Language Development in the School Years*. Cambridge, MA: Brookline Books.
- Durkin, Kevin, Robert D. Crowther, and Beatrice Shire. 1986. "Children's processing of polysemous vocabulary." In *Language Development in the School Years*, edited by Kevin Durkin. Cambridge, MA: Brookline Books.
- Galda, Lee. 1984. "Narrative Competence: Play, Storytelling, and Story Comprehension." In *The Development of Oral and Written Language in Social Contexts*, edited by Anthony Pellegrini and Thomas Yawkey. Norwood, NJ: Ablex Publishing.
- Klavans, Judith, and Min-Yen Kan. 1998. "Role of Verbs in Document Analysis." In *The Proceedings of COLING/ACL 98*. Montreal, Canada.

- Manaris, Bill, and Brian Slator. 1996. "Interactive Natural Language Processing: Building on Success." *Computer* 29. pp. 28-32.
- Moorman, Kenneth, and Ashwin Ram. 1994. "A Model of Creative Understanding." In *Proceedings of the 12th National AAAI Conference*. Seattle, WA.
- Murray, Janet. 1991. "Anatomy of a New Medium: Literary and Pedagogic Uses of Advanced Linguistics Computer Structures." *Computers and the Humanities* 25. pp. 1-14.
- Papka, Ron and James Allan. 1998. "Document Classification using Multiword Features." In *Proceedings of the 7th International Conference on Information and Knowledge Management*. Bethesda, MD.
- Porter, M.F. 1980. "An Algorithm For Suffix Stripping," *Program* 14. pp. 130-137.
- Resnik, Philip. 1995. "Disambiguating Noun Groupings with Respect to WordNet Senses." In *ACL Third Workshop on Very Large Corpora*.
- Robinson, Elizabeth J., and Stephen J. Whittaker. 1986. "Learning about Verbal Referential Communication in the Early School Years." In *Language Development in the School Years*, edited by Kevin Durkin. Cambridge, MA: Brookline Books.
- Sussna, Michael. 1993. "Word Sense Disambiguation for Free-Text Indexing Using a Massive Semantic Network." In *Proceedings of the Second International Conference on Information and Knowledge Management (CIKM-93)*. Arlington, VA.
- Tapanainen, Pasi, and Timo Jarvinen. 1994. "Syntactic Analysis of Natural Language Using Linguistic Rules and Corpus-Based Patterns." In *The 15th International*

Conference on Computational Linguistics Proceedings (COLING 94). Kyoto,
Japan.

Tombros, Anastasios, and Mark Sanderson. 1998. "Advantages of Query Biased
Summaries in Information Retrieval." *SIGIR Forum*. pp. 2-10.

Appendix A: The First Survey

Below is a set of five stories. Each story is followed by two or three fables. Please read each story and its associated fables, and then rank the fables according to how relevant they are to the original story (where 1 means most relevant). Then briefly describe why you ranked the fables the way you did.

1. Original story:

I was in high school. I had been playing on the tennis team all year long. My partner and I made it to the district championships. We need to win that day in order to go on to the statewide tournament. But I was sick. Very sick. I kept throwing up and so I thought about not going to the district tournament. My partner was understanding, but he wanted to play at state. So I went. I was sick through the whole tournament. Between games I would leave the court and vomit. Each point won was very difficult. But I stayed with it because my partner kept encouraging me. At the end of the day we were the winners and so we went on to the state tournament.

Associated fables:

__A. The Shepherd's Boy

There was once a young Shepherd Boy who tended his sheep at the foot of a mountain near a dark forest. It was rather lonely for him all day, so he thought upon a plan by which he could get a little company and some excitement. He rushed down towards the village calling out "Wolf, Wolf," and the villagers came out to meet him, and some of them stopped with him for a considerable time. This pleased the boy so much that a few days afterwards he tried the same trick, and again the villagers came to his help. But shortly after this a Wolf actually did come out from the forest, and began to worry the sheep, and the boy of course cried out "Wolf, Wolf," still louder than before. But this time the villagers, who had been fooled twice before, thought the boy was again deceiving them, and nobody stirred to come to his help. So the Wolf made a good meal off the boy's flock, and when the boy complained, the wise man of the village said:

"A liar will not be believed, even when he speaks the truth."

__B. The Labourer and the Nightingale

A Labourer lay listening to a Nightingale's song throughout the summer night. So pleased was he with it that the next night he set a trap for it and captured it. "Now that I have caught thee," he cried, "thou shalt always sing to me."

"We Nightingales never sing in a cage." said the bird.

"Then I'll eat thee." said the Labourer. "I have always heard say that a nightingale on toast is dainty morsel."

"Nay, kill me not," said the Nightingale; "but let me free, and I'll tell thee three things far better worth than my poor body." The Labourer let him loose, and he flew up to a branch of a tree and said: "Never believe a captive's promise; that's one thing. Then again: Keep what you have. And third piece of advice is: Sorrow not over what is lost forever." Then the song-bird flew away.

C. The Frog and the Ox

"Oh Father," said a little Frog to the big one sitting by the side of a pool, "I have seen such a terrible monster! It was as big as a mountain, with horns on its head, and a long tail, and it had hoofs divided in two."

"Tush, child, tush," said the old Frog, "that was only Farmer White's Ox. It isn't so big either; he may be a little bit taller than I, but I could easily make myself quite as broad; just you see." So he blew himself out, and blew himself out, and blew himself out. "Was he as big as that?" asked he.

"Oh, much bigger than that," said the young Frog.

Again the old one blew himself out, and asked the young one if the Ox was as big as that.

"Bigger, father, bigger," was the reply. So the Frog took a deep breath, and blew and blew and blew, and swelled and swelled and swelled. And then he said: "I'm sure the Ox is not as big as..." But at this moment he burst.

Self-conceit may lead to self-destruction.

Why did you rank these stories the way you did?

2. Original story:

I remember one day this summer I was cooking some hot chicken. The recipe demanded that I heat the oil to a very high temperature and empty out all the contents of a mixed spice packet into it. Well I did just that and started stirring. All of a sudden I realized that a splatter of oil was coming straight for my eye and before I could duck or close my eyes it landed straight in my left eye! Not only was the oil hot in temperature but also was mixed with spices. I tried to open my eye but only in vain. Then my chemistry laboratory training came in handy and I rushed to the sink and flushed my eye under the tap for about 10 minutes like they had taught us in lab. Finally after 10 minutes I was able to open my eye and was relieved that there was nothing serious.

Associated fables:

__A. The Frog and the Ox

"Oh Father," said a little Frog to the big one sitting by the side of a pool, "I have seen such a terrible monster! It was as big as a mountain, with horns on its head, and a long tail, and it had hoofs divided in two."

"Tush, child, tush," said the old Frog, "that was only Farmer White's Ox. It isn't so big either; he may be a little bit taller than I, but I could easily make myself quite as broad; just you see." So he blew himself out, and blew himself out, and blew himself out. "Was he as big as that?" asked he.

"Oh, much bigger than that," said the young Frog.

Again the old one blew himself out, and asked the young one if the Ox was as big as that.

"Bigger, father, bigger," was the reply. So the Frog took a deep breath, and blew and blew and blew, and swelled and swelled and swelled. And then he said: "I'm sure the Ox is not as big as..." But at this moment he burst.

Self-conceit may lead to self-destruction.

__B. The Hart in the Ox-Stall

A Hart hotly pursued by the hounds fled for refuge into an ox-stall, and buried itself in a truss of hay, leaving nothing to be seen but the tips of his horns. Soon after the Hunters came up and asked if any one had seen the Hart. The stable boys, who had been resting after their dinner, looked round, but could see nothing, and the Hunters went away. Shortly afterwards the master came in, and looking round, saw that something unusual had taken place. He pointed to the truss of hay and said: "What are those two curious things sticking out of the hay?" And when the stable boys came to look they discovered the Hart, and soon made an end of him. He thus learnt that

Nothing escapes the master's eye.

__C. The Cat-Maiden

The gods were once disputing whether it was possible for a living being to change its nature. Jupiter said "Yes," but Venus said "No." So, to try the question, Jupiter turned a Cat into a Maiden, and gave her to a young man for a wife. The wedding was duly performed and the young couple sat down to the wedding-feast. "See," said Jupiter, to Venus, "how becomingly she behaves. Who could tell that yesterday she was but a Cat? Surely her nature is changed?"

"Wait a minute," replied Venus, and let loose a mouse into the room. No sooner did the bride see this than she jumped up from her seat and tried to pounce upon the mouse. "Ah, you see," said Venus,

"Nature will out."

Why did you rank these stories the way you did?

3. Original story:

Transferring from Vanderbilt to MIT was a particularly difficult decision for me. After finishing my first year, I began to have doubts as to whether Vanderbilt could help me reach my full academic potential. At the same time, though, I had spent a year settling down and making friends. I was now suddenly considering completely uprooting myself and starting over. My dilemma was thus choosing between my education and the life I had made for myself at Vanderbilt. Initially, my emotional attachment to my friends outweighed any benefits of transferring. Fortunately, my friends turned out to be my most valuable source of advice. I think to some extent they sensed my dissatisfaction with the academics at Vanderbilt, and they thus encouraged me to find a more challenging environment. More importantly though, they helped to dispel my greatest fear that by transferring to MIT I would somehow be losing them. By overcoming that concern, I grew confident that joining MIT was the right decision.

Once I reached MIT I had to leave this support behind and fend for myself. I soon found that MIT prides itself on giving its students a great deal of independence. This policy, however, tends to leave transfer students hanging in the balance. I survived my first semester at MIT thanks solely to my own persistence and force of will. I had to fight for almost everything from transfer credit to housing. In addition, I had to accomplish this while dealing with my classes and schoolwork. My most important challenge, though, was to settle in socially. I knew I would not survive long without the same support base I enjoyed at Vanderbilt. Luckily, everybody I met was very accepting and it was not long before I felt at home. Thanks to my friends at Vanderbilt and MIT and my own determination, changing schools did not become the nightmare I envisioned, but instead has been one of my most rewarding experiences.

Associated fables:

___A. The Frog and the Ox

"Oh Father," said a little Frog to the big one sitting by the side of a pool, "I have seen such a terrible monster! It was as big as a mountain, with horns on its head, and a long tail, and it had hoofs divided in two."

"Tush, child, tush," said the old Frog, "that was only Farmer White's Ox. It isn't so big either; he may be a little bit taller than I, but I could easily make myself quite as broad; just you see." So he blew himself out, and blew himself out, and blew himself out. "Was he as big as that?" asked he.

"Oh, much bigger than that," said the young Frog.

Again the old one blew himself out, and asked the young one if the Ox was as big as that.

"Bigger, father, bigger," was the reply. So the Frog took a deep breath, and blew and blew and blew, and swelled and swelled and swelled. And then he said: "I'm sure the Ox is not as big as..." But at this moment he burst.

Self-conceit may lead to self-destruction.

B. The Hare With Many Friends

A Hare was very popular with the other beasts who all claimed to be her friends. But one day she heard the hounds approaching and hoped to escape them by the aid of her many Friends. So, she went to the horse, and asked him to carry her away from the hounds on his back. But he declined, stating that he had important work to do for his master. "He felt sure," he said, "that all her other friends would come to her assistance." She then applied to the bull, and hoped that he would repel the hounds with his horns. The bull replied: "I am very sorry, but I have an appointment with a lady; but I feel sure that our friend the goat will do what you want." The goat, however, feared that his back might do her some harm if he took her upon it. The ram, he felt sure, was the proper friend to apply to. So she went to the ram and told him the case. The ram replied: "Another time, my dear friend. I do not like to interfere on the present occasion, as hounds have been known to eat sheep as well as hares." The Hare then applied, as a last hope, to the calf, who regretted that he was unable to help her, as he did not like to take the responsibility upon himself, as so many older persons than himself had declined the task. . By this time the hounds were quite near, and the Hare took to her heels and luckily escaped.

He that has many friends, has no friends.

Why did you rank these stories the way you did?

4. Original story:

The water was cold and clear. I had my snorkeling gear ready to go. It didn't look too difficult, but then again, looks could be deceiving. I dove into the water, not knowing what to expect. The current was strong, so I had to make an effort to stay in one general area near the shore. When I finally looked through the snorkeling mask into the water, I could not believe what I saw.

It was like watching an episode from Jacques Cousteau. The water was crystal clear and the my breathing was controlled. Even though I had to stay near the surface of the water, I could see everything to the bottom of the ocean. Schools of fish swam by my face as if I wasn't even there. The plankton was shining beautifully in the sunny water. All in all, I uncovered a whole new world that I had never known existed.

Associated fables:

__A. The Fox and the Mask

A Fox had by some means got into the store-room of a theatre. Suddenly he observed a face glaring down on him and began to be very frightened; but looking more closely he found it was only a Mask such as actors use to put over their face. "Ah," said the Fox, "you look very fine; it is a pity you have not got any brains."

Outside show is a poor substitute for inner worth.

__B. The Crow and the Pitcher

A Crow, half-dead with thirst, came upon a Pitcher, which had once been full of water; but when the Crow put its beak into the mouth of the Pitcher he found that only very little water was left in it, and that he could not reach far enough down to get at it. He tried, and he tried, but at last had to give up in despair. Then a thought came to him, and he took a pebble and dropped it into the Pitcher.

Then he took another pebble and dropped it into the Pitcher.

Then he took another pebble and dropped that into the Pitcher.

Then he took another pebble and dropped that into the Pitcher.

Then he took another pebble and dropped that into the Pitcher.

Then he took another pebble and dropped that into the Pitcher.

At last, at last, he saw the water mount up near him, and after casting in a few more pebbles he was able to quench his thirst and save his life.

Little by little does the trick.

__C. The Frog and the Ox

"Oh Father," said a little Frog to the big one sitting by the side of a pool, "I have seen such a terrible monster! It was as big as a mountain, with horns on its head, and a long tail, and it had hoofs divided in two."

"Tush, child, tush," said the old Frog, "that was only Farmer White's Ox. It isn't so big either; he may be a little bit taller than I, but I could easily make myself quite as broad; just you see." So he blew himself out, and blew himself out, and blew himself out. "Was he as big as that?" asked he.

"Oh, much bigger than that," said the young Frog.

Again the old one blew himself out, and asked the young one if the Ox was as big as that.

"Bigger, father, bigger," was the reply. So the Frog took a deep breath, and blew and blew and blew, and swelled and swelled and swelled. And then he said: "I'm sure the Ox is not as big as..." But at this moment he burst.

Self-conceit may lead to self-destruction.

Why did you rank these stories the way you did?

5. Original story:

The Gate

Three students seeking Admission to the University arrive at its open gate. The Master greets them there. To each he gives the same instruction: "To be Admitted, you must pass through this gate, twice."

The first student walks into the grounds through the gate, then out again. "You have arrived where you belong," says the Master. "You are not Admitted."

The second student walks through the gate, then climbs the wall, circles the University grounds, and enters through it again. "In some versions, he circles the entire world, and returns an old man.) "You are incapable of learning more than you already know," says the Master. "You are not Admitted." The student is enraged, and attacks the Master, who kills him.

The third student, on hearing the Master's demand, turns without a word and walks away. "Wait," says the Master. "Why are you leaving?"

"The task is impossible," says the student. "Pass through that gate twice. If I pass through it once, it will not be the same gate, and I will not be I."

"You are Admitted but not Graduated," says the Master. "You err in believing that you can pass through a gate once."

Associated fables:

__A. The Bat, the Birds, and the Beasts

A great conflict was about to come off between the Birds and the Beasts. When the two armies were collected together the Bat hesitated which to join. The Birds that passed his perch said: "Come with us"; but he said: "I am a Beast." Later on, some Beasts who were passing underneath him looked up and said: "Come with us"; but he said: "I am a Bird." Luckily at the last moment peace was made, and no battle took place, so the Bat came to the Birds and wished to join in the rejoicings, but they all turned against him and he had to fly away. He then went to the Beasts, but soon had to beat a retreat, or else they would have torn him to pieces. "Ah," said the Bat, "I see now,

"He that is neither one thing nor the other has no friends."

B. The Fisher

A Fisher once took his bagpipes to the bank of a river, and played upon them with the hope of making the fish rise; but never a one put his nose out of the water. So he cast his net into the river and soon drew it forth filled with fish. Then he took his bagpipes again, and, as he played, the fish leapt up in the net. "Ah, you dance now when I play," said he.

"Yes," said an old Fish:

"When you are in a man's power you must do as he bids you."

Why did you rank these stories the way you did?

Appendix B: The Second Survey

Below is a set of five stories. Each story is followed by between two and four fables. Please read each story and its associated fables, and then rank the fables according to how relevant they are to the original story (where 1 means most relevant). Then briefly describe why you ranked the fables the way you did.

1. Original story:

This past break, I had a dance performance. I hadn't been able to practice much in the preceding weeks because I was busy with college work, but the week before my performance I went to dance class every day. I had the fortune to be dancing with my friend, which makes it much more fun, but much harder because we needed to coordinate. To make things harder, my friend is also at college and our breaks did not coincide. So we were only able to practice on the weekend before and the day before and the day of the show.

When we went on stage, I could feel my lack of practice. I became exhausted during the second dance itself, and for the first time just stood still on stage having forgotten a step. In other places, my mistakes stood out since the audience could tell if my dancemate and I were doing different steps.

I realized after that dance that I needed to pay attention to the wise wisdom of my dance teacher and mother and drink something to get my energy up. With a vengeance I performed the next dance, determined to get it right. My mistakes were very few throughout... and then we got to the end of the piece. I had to spin around the entire stage-- a step I was yet to do a single time without slipping. I started, and began spinning much faster than I usually do, and completed it without a flaw. To my surprise the audience started clapping for the spins, and I recovered my confidence. My friend and I finished the rest of the concert in pretty good style, and enjoyed the performance despite the flaws, since dancing is always fun no matter what for us.

Associated fables:

A. The Frogs Desiring a King

The Frogs were living as happy as could be in a marshy swamp that just suited them; they went splashing about caring for nobody and nobody troubling with them. But some of them thought that this was not right, that they should have a king and a proper constitution, so they determined to send up a petition to Jove to give them what they wanted. "Mighty Jove," they cried, "send unto us a king that will rule over us and keep us in order." Jove laughed at their croaking, and threw down into the swamp a huge Log, which came down and splash into the swamp. The Frogs were frightened out of their lives by the commotion made in their midst, and all rushed to the bank to look at the horrible monster; but after a time, seeing that it did not move, one or two of the boldest of them ventured out towards the Log, and even dared to touch it; still it did not move. Then the greatest hero of the Frogs jumped upon the Log and commenced dancing up and down upon it, thereupon all the Frogs came and did the same; and for some time

the Frogs went about their business every day without taking the slightest notice of their new King Log lying in their midst. But this did not suit them, so they sent another petition to Jove, and said to him, "We want a real king; one that will really rule over us." Now this made Jove angry, so he sent among them a big Stork that soon set to work gobbling them all up. Then the Frogs repented when too late.

Better no rule than cruel rule.

B. The Hare With Many Friends

A Hare was very popular with the other beasts who all claimed to be her friends. But one day she heard the hounds approaching and hoped to escape them by the aid of her many Friends. So, she went to the horse, and asked him to carry her away from the hounds on his back. But he declined, stating that he had important work to do for his master. "He felt sure," he said, "that all her other friends would come to her assistance." She then applied to the bull, and hoped that he would repel the hounds with his horns. The bull replied: "I am very sorry, but I have an appointment with a lady; but I feel sure that our friend the goat will do what you want." The goat, however, feared that his back might do her some harm if he took her upon it. The ram, he felt sure, was the proper friend to apply to. So she went to the ram and told him the case. The ram replied: "Another time, my dear friend. I do not like to interfere on the present occasion, as hounds have been known to eat sheep as well as hares." The Hare then applied, as a last hope, to the calf, who regretted that he was unable to help her, as he did not like to take the responsibility upon himself, as so many older persons than himself had declined the task. . By this time the hounds were quite near, and the Hare took to her heels and luckily escaped.

He that has many friends, has no friends.

C. The Bat, the Birds, and the Beasts

A great conflict was about to come off between the Birds and the Beasts. When the two armies were collected together the Bat hesitated which to join. The Birds that passed his perch said: "Come with us"; but he said: "I am a Beast." Later on, some Beasts who were passing underneath him looked up and said: "Come with us"; but he said: "I am a Bird." Luckily at the last moment peace was made, and no battle took place, so the Bat came to the Birds and wished to join in the rejoicings, but they all turned against him and he had to fly away. He then went to the Beasts, but soon had to beat a retreat, or else they would have torn him to pieces. "Ah," said the Bat, "I see now,

"He that is neither one thing nor the other has no friends."

Why did you rank these stories the way you did?

2. Original story:

I remember one day this summer I was cooking some hot chicken. The recipe demanded that I heat the oil to a very high temperature and empty out all the contents of a mixed spice packet into it. Well I did just that and started stirring. All of a sudden I realized that a splatter of oil was coming straight for my eye and before I could duck or close my eyes it landed straight in my left eye! Not only was the oil hot in temperature but also was mixed with spices. I tried to open my eye but only in vain. Then my chemistry laboratory training came in handy and I rushed to the sink and flushed my eye under the tap for about 10 minutes like they had taught us in lab. Finally after 10 minutes I was able to open my eye and was relieved that there was nothing serious.

Associated fables:

__A. The One-Eyed Doe

A Doe had had the misfortune to lose one of her eyes, and could not see any one approaching her on that side. So to avoid any danger she always used to feed on a high cliff near the sea, with her sound eye looking towards the land. By this means she could see whenever the hunters approached her on land, and often escaped by this means. But the hunters found out that she was blind of one eye, and hiring a boat rowed under the cliff where she used to feed and shot her from the sea. "Ah," cried she with her dying voice,

"You cannot escape your fate."

__B. The Cat-Maiden

The gods were once disputing whether it was possible for a living being to change its nature. Jupiter said "Yes," but Venus said "No." So, to try the question, Jupiter turned a Cat into a Maiden, and gave her to a young man for a wife. The wedding was duly performed and the young couple sat down to the wedding-feast. "See," said Jupiter, to Venus, "how becomingly she behaves. Who could tell that yesterday she was but a Cat? Surely her nature is changed?"

"Wait a minute," replied Venus, and let loose a mouse into the room. No sooner did the bride see this than she jumped up from her seat and tried to pounce upon the mouse. "Ah, you see," said Venus,

"Nature will out."

__C. The Old Woman and the Wine-Jar

You must know that sometimes old women like a glass of wine. One of this sort once found a Wine-jar lying in the road, and eagerly went up to it hoping to find it full. But when she took it up she found that all the wine had been drunk out of it. Still she took a long sniff at the mouth of the Jar. "Ah," she cried,

"What memories cling 'round the instruments of our pleasure."

Why did you rank these stories the way you did?

3. Original story:

Transferring from Vanderbilt to MIT was a particularly difficult decision for me. After finishing my first year, I began to have doubts as to whether Vanderbilt could help me reach my full academic potential. At the same time, though, I had spent a year settling down and making friends. I was now suddenly considering completely uprooting myself and starting over. My dilemma was thus choosing between my education and the life I had made for myself at Vanderbilt. Initially, my emotional attachment to my friends outweighed any benefits of transferring. Fortunately, my friends turned out to be my most valuable source of advice. I think to some extent they sensed my dissatisfaction with the academics at Vanderbilt, and they thus encouraged me to find a more challenging environment. More importantly though, they helped to dispel my greatest fear that by transferring to MIT I would somehow be losing them. By overcoming that concern, I grew confident that joining MIT was the right decision.

Once I reached MIT I had to leave this support behind and fend for myself. I soon found that MIT prides itself on giving its students a great deal of independence. This policy, however, tends to leave transfer students hanging in the balance. I survived my first semester at MIT thanks solely to my own persistence and force of will. I had to fight for almost everything from transfer credit to housing. In addition, I had to accomplish this while dealing with my classes and schoolwork. My most important challenge, though, was to settle in socially. I knew I would not survive long without the same support base I enjoyed at Vanderbilt. Luckily, everybody I met was very accepting and it was not long before I felt at home. Thanks to my friends at Vanderbilt and MIT and my own determination, changing schools did not become the nightmare I envisioned, but instead has been one of my most rewarding experiences.

Associated fables:

__A. The Lion and the Mouse

Once when a Lion was asleep a little Mouse began running up and down upon him; this soon wakened the Lion, who placed his huge paw upon him, and opened his big jaws to swallow him. "Pardon, O King," cried the little Mouse: "forgive me this time, I shall never forget it: who knows but what I may be able to do you a turn some of these days?" The Lion was so tickled at the idea of the Mouse being able to help him, that he lifted up his paw and let him go. Some time after the Lion was caught in a trap, and the hunters who desired to carry him alive to the King, tied him to a tree while they went in search of a waggon to carry him on. Just then the little Mouse happened to pass by, and seeing the sad plight in which the Lion was, went up to him and soon gnawed away the ropes that bound the King of the Beasts. "Was I not right?" said the little Mouse.

Little friends may prove great friends.

__B. The Hare With Many Friends

A Hare was very popular with the other beasts who all claimed to be her friends. But one day she heard the hounds approaching and hoped to escape them by the aid of her many Friends. So, she went to the horse, and asked him to carry her away from the hounds on his back. But he declined, stating that he had important work to do for his master. "He felt sure," he said, "that all her other friends would come to her assistance." She then applied to the bull, and hoped that he would repel the hounds with his horns. The bull replied: "I am very sorry, but I have an appointment with a lady; but I feel sure that our friend the goat will do what you want." The goat, however, feared that his back might do her some harm if he took her upon it. The ram, he felt sure, was the proper friend to apply to. So she went to the ram and told him the case. The ram replied: "Another time, my dear friend. I do not like to interfere on the present occasion, as hounds have been known to eat sheep as well as hares." The Hare then applied, as a last hope, to the calf, who regretted that he was unable to help her, as he did not like to take the responsibility upon himself, as so many older persons than himself had declined the task. . By this time the hounds were quite near, and the Hare took to her heels and luckily escaped.

He that has many friends, has no friends.

Why did you rank these stories the way you did?

4. Original story:

And now I must go back to the commencement of last month, when we were daily expecting to leave Fayal. We were detained some days partly by the weather, and partly on account of an American ship which had to be condemned, and her crew and part of her cargo were to return to America in the Harbinger. At last everything was ready, and the day came, the 9th of Oct., and a Friday, an inauspicious day for the commencement of a voyage. We did not leave till the afternoon, and the parting was painful for all, particularly for Sarah, who felt, in losing her mother, that she might not again see her for a very long time. I felt exceedingly at parting from friends, who had treated me with such unvarying kindness, and who are all so charming, that no one who knows them, can help loving them, but I left with the hope of meeting again, for Mr. and Mrs. Dabney, with Fanny, talk of visiting the states next spring. Kate also, will then return. They quay was crowded with people come to bid us farewell, as you know is the custom in the Azores, and made me think, Juliet, of our departure from St. Michael's. In the midst of our adieux, we were summoned away to hasten into the boat, there being much difficulty in embarking on account of the surf, which was very high, so much as to prevent any ladies from accompanying us on board. Poor Kate, felt dreadfully at parting with her Mother, she had never before been separated from her parents. Though there was much swell, there was not sufficient breeze stirring for us to leave the bay until it had become quite dark; we sat on the deck, talking to Mr. Dabney, his brother Fredric, and John, who remained with us till we were fairly under weigh, and when their boat distanced the ship, we all felt, that we had in truth, bid adieu to Fayal for some time to come. The next thought, was making ourselves comfortable before being victimized by sea-sickness.

Associated fables:

___A. The Wolf and the Crane

A Wolf had been gorging on an animal he had killed, when suddenly a small bone in the meat stuck in his throat and he could not swallow it. He soon felt terrible pain in his throat, and ran up and down groaning and groaning and seeking for something to relieve the pain. He tried to induce every one he met to remove the bone. "I would give anything," said he, "if you would take it out." At last the Crane agreed to try, and told the Wolf to lie on his side and open his jaws as wide as he could. Then the Crane put its long neck down the Wolf's throat, and with its beak loosened the bone, till at last it got it out.

"Will you kindly give me the reward you promised?" said the Crane.

The Wolf grinned and showed his teeth and said: "Be content. You have put your head inside a Wolf's mouth and taken it out again in safety; that ought to be reward enough for you."

Gratitude and greed go not together.

__B. The Woodman and the Serpent

One wintry day a Woodman was tramping home from his work when he saw something black lying on the snow. When he came closer he saw it was a Serpent to all appearance dead. But he took it up and put it in his bosom to warm while he hurried home. As soon as he got indoors he put the Serpent down on the hearth before the fire. The children watched it and saw it slowly come to life again. Then one of them stooped down to stroke it, but the Serpent raised its head and put out its fangs and was about to sting the child to death. So the Woodman seized his axe, and with one stroke cut the Serpent in two. "Ah," said he,

"No gratitude from the wicked."

__C. Hercules and the Waggoner

A Waggoner was once driving a heavy load along a very muddy way. At last he came to a part of the road where the wheels sank half-way into the mire, and the more the horses pulled, the deeper sank the wheels. So the Waggoner threw down his whip, and knelt down and prayed to Hercules the Strong. "O Hercules, help me in this my hour of distress," quoth he. But Hercules appeared to him, and said:

"Tut, man, don't sprawl there. Get up and put your shoulder to the wheel."

The gods help them that help themselves.

__D. The Hare With Many Friends

A Hare was very popular with the other beasts who all claimed to be her friends. But one day she heard the hounds approaching and hoped to escape them by the aid of her many Friends. So, she went to the horse, and asked him to carry her away from the hounds on his back. But he declined, stating that he had important work to do for his master. "He felt sure," he said, "that all her other friends would come to her assistance." She then applied to the bull, and hoped that he would repel the hounds with his horns. The bull replied: "I am very sorry, but I have an appointment with a lady; but I feel sure that our friend the goat will do what you want." The goat, however, feared that his back might do her some harm if he took her upon it. The ram, he felt sure, was the proper friend to apply to. So she went to the ram and told him the case. The ram replied: "Another time, my dear friend. I do not like to interfere on the present occasion, as hounds have been known to eat sheep as well as hares." The Hare then applied, as a last hope, to the calf, who regretted that he was unable to help her, as he did not like to take the responsibility upon himself, as so many older persons than himself had declined the task. . By this time the hounds were quite near, and the Hare took to her heels and luckily escaped.

He that has many friends, has no friends.

Why did you rank these stories the way you did?

5. Original story:

The Gate

Three students seeking Admission to the University arrive at its open gate. The Master greets them there. To each he gives the same instruction: "To be Admitted, you must pass through this gate, twice."

The first student walks into the grounds through the gate, then out again. "You have arrived where you belong," says the Master. "You are not Admitted."

The second student walks through the gate, then climbs the wall, circles the University grounds, and enters through it again. "In some versions, he circles the entire world, and returns an old man.) "You are incapable of learning more than you already know," says the Master. "You are not Admitted." The student is enraged, and attacks the Master, who kills him.

The third student, on hearing the Master's demand, turns without a word and walks away. "Wait," says the Master. "Why are you leaving?"

"The task is impossible," says the student. "Pass through that gate twice. If I pass through it once, it will not be the same gate, and I will not be I."

"You are Admitted but not Graduated," says the Master. "You err in believing that you can pass through a gate once."

Associated fables:

__A. The Fisher

A Fisher once took his bagpipes to the bank of a river, and played upon them with the hope of making the fish rise; but never a one put his nose out of the water. So he cast his net into the river and soon drew it forth filled with fish. Then he took his bagpipes again, and, as he played, the fish leapt up in the net. "Ah, you dance now when I play," said he.

"Yes," said an old Fish:

"When you are in a man's power you must do as he bids you."

__B. The Bat, the Birds, and the Beasts

A great conflict was about to come off between the Birds and the Beasts. When the two armies were collected together the Bat hesitated which to join. The Birds

that passed his perch said: "Come with us"; but he said: "I am a Beast." Later on, some Beasts who were passing underneath him looked up and said: "Come with us"; but he said: "I am a Bird." Luckily at the last moment peace was made, and no battle took place, so the Bat came to the Birds and wished to join in the rejoicings, but they all turned against him and he had to fly away. He then went to the Beasts, but soon had to beat a retreat, or else they would have torn him to pieces. "Ah," said the Bat, "I see now,

"He that is neither one thing nor the other has no friends."

Why did you rank these stories the way you did?

Appendix C: Compilation Results

The following story and associated lists of words represent the intermediate stages of compilation for one of Aesop's stories ("The Lion and the Mouse"), under the stemming middle-weight algorithm. The first section is the text of the story. The next section is the contents of the keyword file for that story, generated after the user inputs keywords, their parts of speech, their senses, and their importance. The third section is the expanded file generated by WordNet expansion, consisting of the original keywords plus their synonyms, hypernyms, and hyponyms. The fourth section is the list of verbs extracted from the story. Finally, the fifth section is the stemmed versions of all those words (expanded keywords and extracted verbs, minus stopwords).

C.1 Story Text

The Lion and the Mouse

Once when a Lion was asleep a little Mouse began running up and down upon him; this soon wakened the Lion, who placed his huge paw upon him, and opened his big jaws to swallow him. "Pardon, O King," cried the little Mouse: "forgive me this time, I shall never forget it: who knows but what I may be able to do you a turn some of these days?" The Lion was so tickled at the idea of the Mouse being able to help him, that he lifted up his paw and let him go. Some time after the Lion was caught in a trap, and the hunters who desired to carry him alive to the King, tied him to a tree while they went in search of a waggon to carry him on. Just then the little Mouse happened to pass by, and seeing the sad plight in which the Lion was, went up to him and soon gnawed away the ropes that bound the King of the Beasts. "Was I not right?" said the little Mouse.

Little friends may prove great friends.

C.2 Keywords

These are the keywords for "The Lion and the Mouse" as chosen by Nick Montfort and myself during the creation of Aesop. The following keywords are associated with a sense number, a part of speech, and an importance, respectively.

little,7,a,9
great,1,a,7
friends,3,n,10
help,1,v,6
need,1,n,7
favor,1,n,4
gratitude,1,n,10
underestimate,3,v,10
prove,1,v,5
reversal,1,n,10
circumstance,1,n,8

C.3 Expanded words

This is the list of expanded keywords (the original keywords plus their synonyms, hyponyms, and hypernyms). The synonyms, hyponyms, and hypernyms of the original keywords (see section C.2) were found using WordNet. These words are associated with a part of speech and a weight, respectively. Since WordNet expansion has been completed, the sense number is no longer needed. Note that the importance values (the weights) no longer vary between 1 and 10, as above, but have been multiplied by a factor depending on each word's relationship to an original keyword (see section 7.2.2.1).

acquaintance,n,20
aid,v,12
alleviate,v,12
appreciativeness,n,20
assist,v,24
attend,v,12
avail,v,12
bootstrap,v,12
bound,v,20
bunkmate,n,20
campmate,n,20
care,v,12
change,n,10
circumstance,n,48
classmate,n,20
connection,n,20
cried,v,20
deficiency,n,14
demand,n,14
desired,v,20
ease,v,12
ensue,v,10

entity,n,20
err,v,10
expedite,v,12
facilitate,v,12
forget,v,20
forgive,v,20
friend,n,20
friends,n,40
gnawed,v,20
gratefulness,n,20
gratitude,n,60
hasten,v,12
help,v,36
hinge,n,16
knows,v,20
lack,n,14
little,a,36
messmate,n,20
mistake,v,10
necessity,n,14
need,n,42
opened,v,20
organism,n,10
pardon,v,20
pickup,n,20
prove,v,30
regress,n,20
regression,n,20
result,v,10
retrogression,n,20
retroversion,n,20
reversal,n,60
reversion,n,20
running,v,20
schoolfellow,n,20
schoolmate,n,20
serve,v,12
slip,v,10
soft,a,18
subserve,v,12
succor,v,12
succour,v,12
swallow,v,20
thankfulness,n,20
tickled,v,20
tied,v,20

turn,n,8
underestimate,v,50
underrate,v,10
wakened,v,20

C.4 Extracted verbs

This is the list of verbs that appear in the text of “The Lion and the Mouse.” The story was parsed by the ENGCG parser, and every word tagged as a verb was added to this list. As these verbs are added to the list of expanded keywords (see section C.3) a part of speech and importance are automatically added.

be
begin
bind
carry
catch
cry
desire
do
forget
forgive
gnaw
go
happen
help
know
let
lift
open
pardon
pass
place
prove
run
say
see
swallow
tickle
tie
waken

C.5 Final list

These are the stemmed versions of the culled list of expanded keywords and verbs. Stopwords have been removed by this stage. The stemming was done by running each word through the Porter stemmer. They are associated with only their weight.

acquaintanc,20
aid,12
allevy,12
apprecy,20
assist,24
attend,12
avail,12
bind,20
bootstrap,12
bunkmat,20
campmat,20
care,12
chang,10
circumst,48
classmat,20
connect,20
cry,20
deficy,14
demand,14
desir,20
eas,12
ensu,10
entity,20
err,10
expedit,12
facilit,12
forget,20
forgiv,20
friend,40
gnaw,20
grateful,20
gratitud,60
hasten,12
help,36
hing,16
know,20
lack,14
littl,36
messmat,20
mistak,10

ne,42
necess,14
open,20
organism,10
pardon,20
pickup,20
prove,30
regress,20
result,10
retrogress,20
retrovers,20
revers,60
run,20
schoolfellow,20
schoolmat,20
serv,12
slip,10
soft,18
subserv,12
succor,12
succour,12
swallow,20
thankful,20
tickl,20
tie,20
turn,8
underestim,50
underrat,10
waken,20

Appendix D: Story matching Results

In this section, an original story is presented, along with two listings of word matches by the matching algorithm. In each listing, an original story word that matches at least one character story is printed, followed by the numbers of the stories it matched and the corresponding weight in parentheses. At the end of the listing, the number of the winning fable is declared.

The original story given here is “Cooking Chicken.” The given listings were produced by the unmodified algorithm and the middle weight inflecting algorithm. For convenience, I have included the two winning fables, as well.

D.1 Original Story

I remember one day this summer I was cooking some hot chicken. The recipe demanded that I heat the oil to a very high temperature and empty out all the contents of a mixed spice packet into it. Well I did just that and started stirring. All of a sudden I realized that a splatter of oil was coming straight for my eye and before I could duck or close my eyes it landed straight in my left eye! Not only was the oil hot in temperature but also was mixed with spices. I tried to open my eye but only in vain. Then my chemistry laboratory training came in handy and I rushed to the sink and flushed my eye under the tap for about 10 minutes like they had taught us in lab. Finally after 10 minutes I was able to open my eye and was relieved that there was nothing serious.

D.2 Unmodified algorithm

remember matches these stories: 81 (60)
summer matches these stories: 36 (8)
cooking matches these stories: 20 (7) 36 (4) 43 (8) 61 (6) 82 (6)
chicken matches these stories: 36 (2) 53 (3)
demanded matches these stories: 46 (8) 60 (5)
heat matches these stories: 17 (16) 56 (8) 60 (8) 77 (8)
temperature matches these stories: 60 (16)
empty matches these stories: 77 (8) 81 (8)
contents matches these stories: 10 (3) 13 (4) 15 (9) 19 (5) 20 (9) 21 (7) 24 (10) 24 (5) 2 (6) 30 (20) 32 (6) 33 (7) 34 (9) 35 (18) 39 (14) 3 (9) 41 (10) 43 (32) 45 (10)

46 (21) 49 (11) 51 (2) 56 (9) 58 (9) 59 (7) 5 (14) 62 (16) 62 (4) 65 (5) 68 (6) 70
 (2) 74 (9) 76 (8) 77 (2) 81 (9) 82 (4) 8 (10)
 mixed matches these stories: 77 (8)
 spice matches these stories: 36 (2)
 stirring matches these stories: 61 (12) 79 (3)
 eye matches these stories: 30 (12) 63 (2) 66 (8)
 duck matches these stories: 36 (2) 53 (3) 66 (6) 67 (16) 75 (3)
 close matches these stories: 77 (8)
 eyes matches these stories: 30 (12) 35 (16) 43 (9) 46 (6) 63 (2) 66 (8)
 landed matches these stories: 77 (8)
 temperature matches these stories: 60 (16)
 mixed matches these stories: 77 (8)
 spices matches these stories: 36 (2)
 tried matches these stories: 24 (20)
 open matches these stories: 52 (14)
 eye matches these stories: 30 (12) 63 (2) 66 (8)
 training matches these stories: 24 (10) 44 (36) 48 (36) 77 (12)
 rushed matches these stories: 26 (14) 68 (4)
 sink matches these stories: 75 (15)
 flushed matches these stories: 77 (12)
 eye matches these stories: 30 (12) 63 (2) 66 (8)
 tap matches these stories: 40 (14) 48 (6) 61 (6)
 taught matches these stories: 44 (36) 48 (36) 77 (4)
 open matches these stories: 52 (14)
 eye matches these stories: 30 (12) 63 (2) 66 (8)
 relieved matches these stories: 6 (10) 77 (4) 80 (4)
 And a winner is ... STORY 77

D.3 Story 77: The Cat-Maiden

The gods were once disputing whether it was possible for a living being to change its nature. Jupiter said "Yes," but Venus said "No." So, to try the question, Jupiter turned a Cat into a Maiden, and gave her to a young man for a wife. The wedding was duly performed and the young couple sat down to the wedding-feast. "See," said Jupiter, to Venus, "how becomingly she behaves. Who could tell that yesterday she was but a Cat? Surely her nature is changed?"

"Wait a minute," replied Venus, and let loose a mouse into the room. No sooner did the bride see this than she jumped up from her seat and tried to pounce upon the mouse. "Ah, you see," said Venus,

"Nature will out."

D.4 Middle weight algorithm with inflection

remember matches these stories: 81 (90) 82 (20)
summer matches these stories: 36 (10)
cooking matches these stories: 36 (4)
hot matches these stories: 34 (2) 62 (6)
heat matches these stories: 17 (4) 56 (12) 60 (8) 77 (8)
temperature matches these stories: 60 (24)
empty matches these stories: 77 (8) 81 (8)
contents matches these stories: 10 (3) 13 (5) 19 (5) 20 (9) 21 (7) 24 (5) 29 (7) 2
(6) 30 (20) 33 (7) 34 (9) 35 (18) 37 (20) 39 (14) 3 (9) 41 (10) 43 (25) 45 (10) 46
(21) 49 (11) 51 (1) 56 (9) 58 (9) 59 (7) 5 (10) 62 (16) 65 (5) 68 (6) 69 (8) 70 (2)
73 (8) 74 (9) 75 (2) 76 (8) 77 (2) 81 (9) 82 (4) 8 (10)
mixed matches these stories: 77 (8)
spice matches these stories: 77 (8)
stirring matches these stories: 43 (20) 44 (8) 79 (3)
eye matches these stories: 30 (18) 66 (20)
duck matches these stories: 66 (6) 67 (4)
close matches these stories: 14 (20) 46 (20) 77 (8)
eyes matches these stories: 30 (18) 35 (16) 66 (20)
landed matches these stories: 77 (8)
hot matches these stories: 34 (2) 62 (6)
temperature matches these stories: 60 (24)
mixed matches these stories: 77 (8)
spices matches these stories: 77 (8)
tried matches these stories: 24 (20) 26 (20) 31 (20) 43 (20) 51 (20) 52 (20) 55 (20)
5 (20) 62 (20) 6 (20) 72 (20) 77 (20)
open matches these stories: 11 (20) 3 (20) 52 (14) 57 (20) 5 (20) 81 (4) 8 (20)
eye matches these stories: 30 (18) 66 (20)
training matches these stories: 31 (20) 44 (36) 48 (36)
rushed matches these stories: 10 (20) 13 (20) 23 (20) 26 (14) 2 (20) 43 (20) 46
(20) 50 (20) 68 (4)
sink matches these stories: 61 (20) 75 (12)
eye matches these stories: 30 (18) 66 (20)
tap matches these stories: 40 (14) 48 (6)
taught matches these stories: 10 (20) 44 (54) 48 (54) 62 (20)
open matches these stories: 11 (20) 3 (20) 52 (14) 57 (20) 5 (20) 81 (4) 8 (20)
eye matches these stories: 30 (18) 66 (20)
relieved matches these stories: 5 (20) 64 (20)
And a winner is ... STORY 81

D.5 Story 81: The Old Woman and the Wine-Jar

You must know that sometimes old women like a glass of wine. One of this sort once found a Wine-jar lying in the road, and eagerly went up to it hoping to find it

full. But when she took it up she found that all the wine had been drunk out of it. Still she took a long sniff at the mouth of the Jar. "Ah," she cried,

"What memories cling 'round the instruments of our pleasure."