Overview

Speech recognition holds the promise of extending the benefits of natural language applications to many communities characterized by the need for hands-free and user-friendly interaction. Realizing this promise presents a unique challenge: accurate rendering of speech to text at the sentence level is critical for natural language technology to correctly process the semantic intent of user communications. Research conducted as part of United States Department of Education SBIR grant H133S080032 demonstrated that natural language technology improved the sentence-level accuracy of the commercial speech recognition software being used as the application's front-end.

People successfully deal with ambiguously-sounding utterances on a normal basis by selecting the one that "makes the most sense" from among various possible interpretations. Speech recognition software has phonetic-processing algorithms to help resolve ambiguously sounding utterances, but lacks the capacity to semantically process, i.e. "make sense of" them. We wanted to explore whether the use of natural language understanding would add value through an extra step of semantic evaluation.

We used the Tridbit natural language technology, as implemented in our JotChat application prototype, to conduct our research. We capitalized on JotChat's ability to understand if a sentence "makes sense" by having it evaluate the speech recognition software's alternative interpretations of a spoken sentence.

Our research clearly indicated that JotChat could deliver significant improvement in sentence-level speech recognition accuracy and thus improve the overall user experience for those employing speech recognition as an application interface.

Executing the research and analyzing the results also gave us valuable guidance for future testing, as well as for research and development in the practical integration of speech recognition with natural language applications.

Methodology

The test involved submitting 30 sentences typical of JotChat interaction to the commercial speech recognition software using a custom interface that could return, for each input sentence, an ordered list of sound-to-text interpretations, called an "N-best" list. We used a speech recognition server created by Nuance, with hosting and interfaces provided by VoVision. Each N-best list contained up to 30 possible interpretations of the input sentence audio, with the first item being the one that the speech recognition algorithms determined was the best interpretation ("Best") and which would be presented to the user in a normal application session. Given the current state of speech recognition, the Best interpretation will not always match what the speaker said ("Target sentence").

We defined three possible speech recognition (SR) outcomes (S1, S2, S3) to organize and evaluate the results. Within each outcome we defined a successful JotChat outcome (JC+) and an unsuccessful one (JC-) based on the correct identification of the Target sentence.

Table 1: Six possible outcomes of speech recognition evaluation

Speech Recognition (SR) outcome	JC+	JC-
S1: SR's Best is Target sentence.	JC confirms SR ranking: JC top-scores SR Best (true positive)	JC does not give SR's Best first top score (false positive) or gives no top score (false negative)
S2: SR's Best is not Target, but Target is on N-best list somewhere in position 2-n.	JC gives Target first top score within N-best list (true positive)	JC does not give Target first top score (false positive) or gives no top score (false negative)

S3: SR did not place Target on N-best list.	JC does not give any N-best list item top score. (true negative)	JC top-scores non-Target (false positive)

To obtain meaningful results from this preliminary test within the constraints of a Phase 1 SBIR, we did the following:

- Limited the test cases to sentences previously determined to be within JotChat's current knowledge set, in order to remove JotChat maturity as a variable in this test.
- Used optimal voice and audio encoding parameters, as determined by pre-test research, to ensure that as many test cases as possible contained the complete Target sentence in the N-best list (S1 and S2).

To execute the test, we submitted each sentence to the speech software and obtained an N-best list for it. We then had JotChat process the N-best lists to score how much each candidate interpretation “made sense”, using a standard letter-grade scale. (See the Technical Discussion for a detailed description of this evaluation and scoring.) Table 2 illustrates three N-best lists after JotChat evaluation. The following describes the format of each list.

The first line of each list contains the Target sentence, which was inserted after the test for reference during analysis and reporting.

The Target is followed by 1 to n three-part items. The first part of each item, prefixed with the item number in the form “<n>”, is a candidate interpretation returned by the speech software and passed to JotChat. The second part is JotChat's response. The third part (“Nbest score =”) is the score JotChat assigned to the interpretation based on its understanding of that interpretation.

The first line of item <1> in each example is the word string that the speech software determined was the Best interpretation of the audio. Now let’s examine each of the examples.

Example 1: the Target is “Alice's phone number is 221-4545” which matches the speech software's Best interpretation (item <1>), so this case is classified as an S1 result. JotChat gave item <1> a top score of “A”, indicating that JotChat's evaluation supports the speech software’s Best, so this case's result classification is further refined to S1 JC+.

Alternatively, had JotChat not scored item <1> as “A” while giving that score to another item (false positive), or if it evaluated no item as “A” (false negative), either outcome would have resulted in a classification of this case as a S1 JC-.

Example 2: the Target is “What is Linda's address?” which does not match the speech software's Best interpretation “plot is Linda's address" (item <1>). However the Target is present in item <3> of the N-best list, so this case is classified as an S2 result. Item <3> is the first item to which JotChat assigned its top score of “A”, so this case's result classification is further refined to S2 JC+.

Alternatively, had JotChat evaluated items <1> or <2> as “A” (false positive), or if it evaluated no item as “A” (false negative), either outcome would have resulted in a classification of this case as a S3 JC-. Note that the classification system only considers JotChat's first top-score. In this hypothetical alternative, an “A” score for either item <1> or <2> generates a false positive even if “A” was correctly assigned to item <3> as well. Our classification system needs an automatic way to determine which “A” to use in the case of multiple top scores, so it uses the first top score.

Example 3: the Target is “What is Paul's phone number?" which is not present anywhere on the N-best list, so this case is classified as an S3 result. JotChat does not give any item its top score, so this case's result classification is further refined to S3 JC+.

Alternatively, had JotChat evaluated any item “A” (false positive), this would have resulted in a classification of this case as a S3 JC-.

Table 2: Three example N-best lists after JotChat evaluation

Example 1: S1 JC+ S1 because SR's Best candidate <1> matches the Target. JC+ because JotChat also correctly identifies Target by scoring <1> as the first A candidate.	Example 2: S2 JC+ S2 because SR's Best candidate <1> does not match the Target and the Target is in the N-best list (<3>). JC+ because JotChat correctly identifies Target by scoring <3> as the first A candidate.	Example 3: S3 JC+ S3 because SR's Best candidate <1> does not match the Target and the Target is not in the N-best list. JC+ because JotChat does not score any A candidates.
Target: Alice's phone number is 221-4545 <1>Alice's phone number is 2214545. OK! NBest score = A <2>Alice's phone number is 2 to 14545. I don't understand. Can you think of another way to say it? NBest score = C4 <3>Alice's phone number is 22145 for 5. I don't understand. Can you think of another way to say it? NBest score = C4 <4>Alice's phone number is 2214545. OK! NBest score = A <5>Alice's phone number is 2 to 145 for 5. I don't understand. Can you think of another way to say it? NBest score = C6 <6>Alice's phone number is 2 to 14545. I don't understand. Can you think of another way to say it? NBest score = C4 <7>Alice's phone number is 22145 for 5. I don't understand. Can you think of another way to say it? NBest score = C4 <8>Alice's phone number is 2 to 145 for 5. I don't understand. Can you think of another way to say it? NBest score = C6	Target: What is Linda's address? <1>plot is Linda's address. I don't understand. Can you think of another way to say it? NBest score = C3 <2>plot is Linda's address. I don't understand. Can you think of another way to say it? NBest score = C3 <3>what is Linda's address. 123 Easy Street, Madison, WI 53705 NBest score = A <4>what is Linda's address. 123 Easy Street, Madison, WI 53705 NBest score = A <5>thought is Linda's address. I don't understand. Can you think of another way to say it? NBest score = C5 <6>plot is Linda's entrance. plot of what is ? NBest score = C3 <7>plot is Linda's entrance. plot of what is ? NBest score = C3 <8>thought is Linda's address. I don't understand. Can you think of another way to say it? NBest score = C5 <9>plot is Linda's attracts. Unknown word 'attracts' NBest score = D (Items 10-30 removed from illustration.)	Target: What is Paul's phone number? <1>what is called phone number. I don't understand. Can you think of another way to say it? NBest score = C2 <2>what is called on member. I do not have that information. Try another question. NBest score = B <3>what is called on number. I do not have that information. Try another question. NBest score = B <4>what is called on number. I do not have that information. Try another question. NBest score = B <5>what is called phone number. I don't understand. Can you think of another way to say it? NBest score = C2 <6>what is called on member. I do not have that information. Try another question. NBest score = B <7>what is called on number. I do not have that information. Try another question. NBest score = B <8>what is called on number. I do not have that information. Try another question. NBest score = B

Results

Table 3: The test results using the classification described in Table 1 and illustrated in Table 2

	S1		S2		S3	
Total	JC+	JC-	JC+	JC-	JC+	JC-
30	15	0	8	1	4	2
	50%	0%	27%	3%	13%	7%

50% of the N-best lists returned by the speech software placed the Target at the top of its N-best list (S1). This represents the success rate baseline in our test for speech recognition unaided by JotChat. JotChat confirmed 100% of these selections (S1 JC+), so there was no degradation of the baseline by applying the JotChat evaluation.

(It is important to note that natural language understanding requires accuracy at the sentence level, so this is how we are measuring successful speech recognition. Conventional methods of assessing speech recognition accuracy often measure at the word level, yielding higher reported accuracy rates.)

In an additional 30% of the cases, the Target was buried within the N-best list (S2). JotChat's evaluation was able to successfully identify the Target in 89% of these (S2 JC+). This drove the overall success rate from 50% unaided (S1) to 77% when aided by JotChat (S1+S2), a significant 53% improvement.

In the remaining 20% of the cases, the N-best list did not contain the Target (S3). JotChat's evaluation confirmed 67% of these (S2 JC+), but generated a false positive in 33% (S2 JC-).

Technical Discussion

The scoring algorithm rates each sentence with a letter score and in some cases a number score. The letter score indicates the following:

A – The sentence made structural sense and if a question, JotChat was able to answer it.

B – The sentence made structural sense, but it was a question JotChat was unable to answer.

C – The sentence did not make structural sense to JotChat.

D – JotChat did not understand the sentence because it contained a word it did not understand.

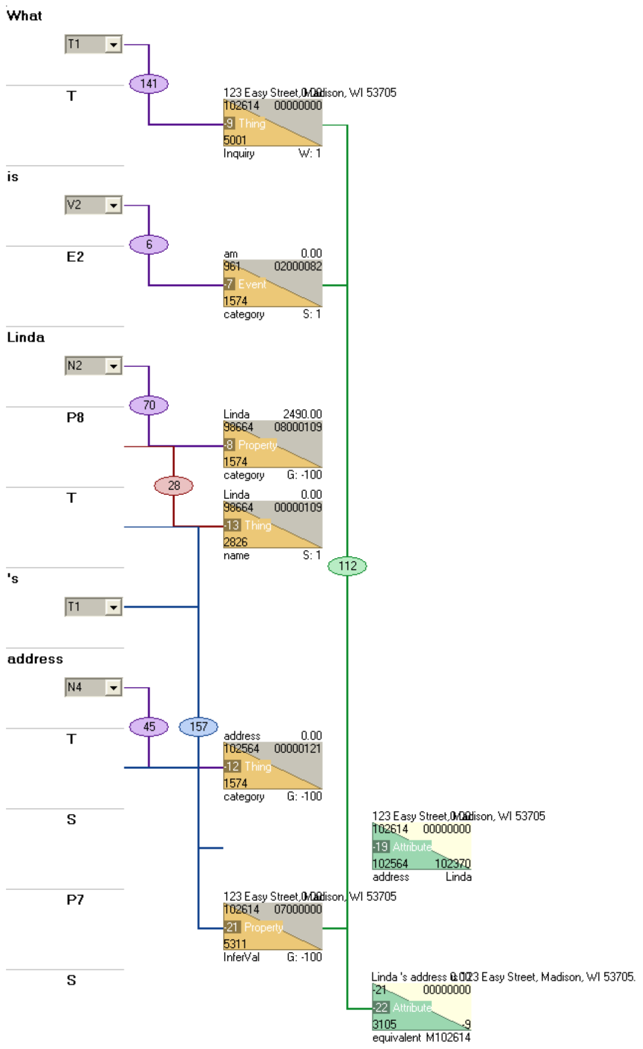
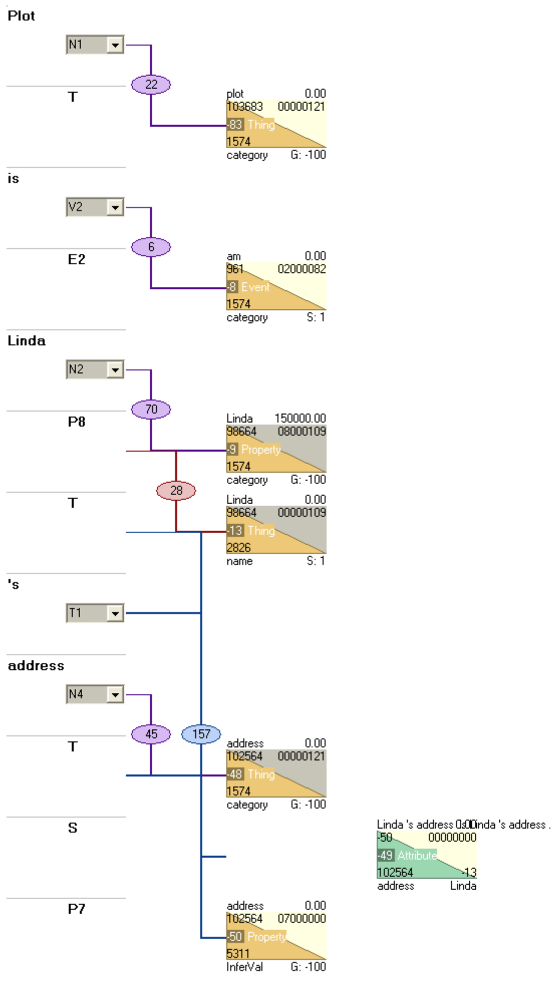
Central to the rating system, and the Tridbit model itself, is the concept of a sentence "making sense." For each sentence the Tridbit engine processes, it attempts to build a meaning structure that represents the information being conveyed by the sentence. It does this by finding patterns and applying the associated transformations and continuing the process until it can tie together all the elements in the sentence.

For most sentences there are many potential patterns that can be processed. The Tridbit engine tries many different sequences of processing patterns to find a sequence that yields a structure that "makes sense." It eliminates possibilities because the basic units of meaning, called tridbits, are highly constrained and must individually be validated to "make sense." The precise method by which this is done is beyond the scope of this document, but is explained in greater detail in:

[Blaedow, K. 2007] *Babble: Simple Conversations With a Computer*. Proceedings of the 2007 Semantic Technology Conference, San Jose, CA. URL = http://www.tridbits.com/docs/simpleconvers.pdf.

One can get a sense of how this works by examining the tridbit representation of a sentence that makes sense vs. one that does not. The table below presents two such examples taken from the N-best lists in table 2. The tridbit diagram shows the sequence of patterns being applied and the resulting tridbits generated. Reading through the explanation provides a sense of how this works without getting too bogged down in the details.

Table 4 – Description and diagram of a sentence that makes sense to JotChat vs. one that does not

Example of a sentence that makes sense to Tridbits	**Example of a sentence that does NOT make sense**
Below is a tridbit diagram of the sentence “What is Linda’s address?” This sentence both makes sense to JotChat and is a question it can answer, so it rates it as an “A”. Each line in the diagram represents a syntax rule. Rules are triggered by matching a pattern of tokens and/or tridbits. The line’s branches connect the tokens and/or tridbits that trigger the rule to its output. The tokens or tridbits are consumed in order to generate a higher-level tridbit. To make sense, the tridbit process must find a set of rules it can apply that consume all but the highest-level tridbits, which represent assertions. In tridbit diagrams, tridbits are shown as the boxed triangles. The green triangles in the right most column represent assert tridbits.	Below is a tridbit diagram of the sentence “Plot is Linda’s address?” This sentence did not make sense to JotChat. It rated it as a “C3” because after applying the rules shown in the diagram, there were three referent tridbits left over that were not consumed. Non-consumption is indicated in the diagram by the box containing the triangle being lit up. The same rules as the “A” rated sentence could be applied to process “Linda’s address”, creating a higher-level tridbit called an *inferval*. Applying the next level of rules fails because the equivalence assert tridbit that would have been generated did not pass the constraints the Tridbit model imposes for this type of assert tridbit to be valid.
 What T1 T 141 123 Easy Street, Madison, WI 53705 0.00 102614 00000000 -9 Thing 5001 Inquiry W: 1 is V2 E2 6 am 0.00 961 02000082 -7 Event 1574 category S: 1 Linda N2 P8 70 Linda 2490.00 98664 08000109 -8 Property 1574 category G: -100 28 T Linda 0.00 98664 00000109 -13 Thing 2826 name S: 1 112 's T1 address N4 T 45 157 address 0.00 102564 00000121 -12 Thing 1574 category G: -100 S 123 Easy Street, Madison, WI 53705 0.00 102614 00000000 -19 Attribute 102564 102370 address Linda P7 123 Easy Street, Madison, WI 53705 0.00 102614 07000000 -21 Property 5311 InferVal G: -100 S Linda 's address 123 Easy Street, Madison, WI 53705. 0.00 -21 00000000 -22 Attribute 3105 -9 equivalent M102614	 Plot N1 T 22 plot 0.00 103683 00000121 -83 Thing 1574 category G: -100 is V2 E2 6 am 0.00 961 02000082 -8 Event 1574 category S: 1 Linda N2 P8 70 Linda 150000.00 98664 08000109 -9 Property 1574 category G: -100 28 T Linda 0.00 98664 00000109 -13 Thing 2826 name S: 1 's T1 address N4 T 45 157 address 0.00 102564 00000121 -48 Thing 1574 category G: -100 S Linda 's address 0.00 Linda 's address . -50 00000000 -49 Attribute 102564 -13 address Linda P7 address 0.00 102564 07000000 -50 Property 5311 InferVal G: -100

The concept of making “structural sense” is a unique characteristic of how Tridbits incorporates constrained structures and a metaphysical model into understanding natural language. This level of making sense can be applied to any sentence, as long as the words have minimal dictionary entries. Not all sentences that make “structural sense” make sense to a human. One example from Table 2, “what is called on member" JotChat scores a "B" even though it does not make sense to us. JotChat does not yet have sufficient deep context knowledge of the concepts involved to know that "on member" is not a clause that makes sense with the verb "call". But in a generic sense, the structure is OK.

JotChat already has some ability to learn deep context knowledge; this is an area where further development will take place. These developments will enhance JotChat's ability to improve speech recognition accuracy.

Conclusions & Future Directions

A. Speech has the potential of being a viable input alternative for JotChat.

The JotChat team continues to see improvements in commercial speech recognition software. For example, the software that we used to integrate speech recognition into our second round of usability testing (Task A5.2) was notably better than the same product one revision earlier, which we evaluated prior to this SBIR.

More importantly, the results clearly indicate that the natural language understanding capability of JotChat can significantly enhance the accuracy of speech recognition at the sentence level, resulting in better JotChat understanding and in a better user experience. The tight coupling of JotChat with speech recognition software can extend JotChat's value to communities for whom keyboarding is not an option (e.g., manual disabilities) and to situations where keyboarding is inappropriate (e.g., shopping).

B. Tridbit technology holds the potential of improving the accuracy of speech input for other applications.

The Tridbit technology that underlies JotChat is a general-purpose engine, so it could, as it matures, be embedded in a range of existing and future applications and devices of an appropriate nature. Improving the speech recognition experience for users of these technologies will greatly expand their utility and reach within disabled and other populations.

The work indicated a number of future directions for continuing to enhance JotChat/speech integration.

- Continue to improve JotChat's algorithms for scoring candidate inputs to further increase its ability to identify the Target sentence (S1 JC+ and S2 JC+) while reducing misidentifications (JC-).
- Feedback JotChat scoring to the training capabilities of the speech recognition software to improve speech recognition accuracy, thereby decreasing the frequency of the Target sentence not being in the N-best list (S3).
- Develop the data capture and reporting software as well as user support mechanisms necessary to move the JotChat/speech usability testing from the lab into the field.