CREATING A LINGUISTIC PLAUSIBILITY DATASET WITH NON-EXPERT ANNOTATORS

Benjamin Lambert, Rita Singh, Bhiksha Raj

School of Computer Science, Carnegie Mellon University, Pittsburgh, PA
belamber@cs.cmu.edu, rsingh@cs.cmu.edu, bhiksha@cs.cmu.edu

ABSTRACT

We describe the creation of a linguistic plausibility dataset that contains annotated examples of language judged to be linguistically plausible, implausible, and every-thing in between. To create the dataset we randomly generate sentences and have them annotated by crowd sourcing over the Amazon Mechanical Turk. Obtaining inter-annotator agreement is a difficult problem because linguistic plausibility is highly subjective. The annotations obtained depend, among other factors, on the manner in which annotators are questioned about the plausibility of sentences. We describe our experiments on posing a number of different questions to the annotators, in order to elicit the responses with greatest agreement, and present several methods for analyzing the resulting responses. The generated dataset and annotations are being made available to public.

Index Terms: Sense. Speech recognition. Mechanical Turk.

1. INTRODUCTION

The following word sequences are from the top-N hypotheses of an automatic recognizer decoding a spoken utterance. Only one of them is correct.

- *people are close population getting older*
- *people are closer population getting older*
- *people are course population getting older*
- *people are clocks population's getting older*

By this time the reader has probably already determined that the fourth hypothesis is the most correct one.

The above example illustrates a problem. The most correct hypothesis was not the top hypothesis of the recognizer. But the human reader was immediately able to recognize it based only on the content of the text, without listening to the audio. Brill [1] has demonstrated that humans are remarkably good at this task. Over a collection of corpora, subjects were able to select a better hypothesis from an N-best list such that word error rate improved by 17%-60% of the maximum achievable over the top hypothesis from the recognizer.

It is generally accepted that the problem of automatic speech recognition could largely be solved if it were possible to computationally replicate this human ability to accurately identify a correct (or most correct) hypothesis from an N-best list, based on their word content. However, arriving at a mechanism that can actually achieve this has been difficult. Various approaches have been proposed for evaluating features that consider sentences as a whole, *e.g* the statistical trigger-pair models of [2], syntactic parse based method of [3] and semantic-coherence based methods [4] among a host of others, but the actual improvement in speech recognition accuracy obtained by employing these methods to select or re-rank word-sequence hypotheses has been minimal.

A large reason for the failure of these approaches is that they are primarily learned from *positive* examples, since the vast majority of textual corpora comprise sentences that people *did* say or write; they are never presented with *negative* examples, sentences that people *would not* say, primarily because a corpus of such text does not exist. Thus while they learn to effectively accept, *i.e.* parse or give high scores to word patterns that are correct, they are not effective at rejecting patterns that are not, particularly when the sentences conform to a generative model such as an N-gram LM. While some authors (*e.g.* [2]) have attempted to simulate *negative* examples by random generation of sentences from an N-gram LM, this is not a satisfactory solution; as we shall see later, humans would consider a large fraction of randomly generated word sequences to be acceptable or meaningful.

What is required, then, is a large corpus of text that includes *both positive and negative* examples, appropriately annotated. *I.e.* we require an corpus which includes both acceptable (as judged by a human), and unacceptable (also as judged by a human) sentences, which have been correctly annotated as such, such that NLP algorithms can be discriminatively trained to accurately distinguish between what is acceptable language and what is wrong. We refer to such a corpus as a ***linguistic-plausibility*** dataset.

This paper deals with the design of such a corpus. We describe the creation of a human-annotated dataset of explicit positive and negative examples of linguistic plausibility. Our approach is to present randomly generated sentences to multiple human annotators and instruct them to tag the sentences as meaningful or not.

"Plausibility" is however very subjective and hard to define, and annotating plausibility is akin to semantic annotation. Even when a clear formalism is specified for the annotation, inter-annotator disagreement tends to be high in semantic annotation tasks [5, 6]. In our context, unlike other annotation tasks, there is no ground truth. Rather, we are simply gathering subjective human judgments about the plausibility of the word sequence as a part of normal human speech or writing. The responses we will elicit from the annotators may be expected to be dependent on various factors relating to the personal context of the annotator, but more importantly *on the manner in which the query about the plausibility of the sentence is posed.* While we may have no control on the former factors, we do have control on the manner in which the question is posed.

Ideally, we would like to pose the query (that asks the annotator if a sentence is plausible) in a manner such that the responses have high inter-annotator agreement, such that positive examples are normal, meaningful sentences and negative examples are fundamentally semantically invalid. However we recognize that such a query is unlikely to exist for most sentences.

In this paper we explore different manners in which this query can be posed in order to obtain maximal inter-annotator agreement. We pose the question about the the plausibility of sentences to the annotators in a number of different ways to determine which one results in the maximal inter-annotator agreement. On a small exploratory set of 100 sentences each of which was tagged by 10 annotators, we find that the best of our queries gives us an average 66% inter-annotator agreement, and that on the same query nearly 50% of the sentences had an inter-annotator agreement of 80% or greater.

Two other key issues about the generation of such a corpus are a) how the sentences are generated, b) who the annotators are. We desire our corpus to be relatively unspecific to any domain, as our intention is to produce a corpus that captures a *general* notion of plausibility (which does exist beyond the extended scope of plausibility within specialized domains). We therefore generate sentences from a trigram LM produced from the Google N-gram data [7], which cover a vast array of topics and are not specific to any domain. We aimn to eventually produce a large corpus of annotated sentences. Obtaining expert taggers to tag such a corpus is both expensive and inefficient. We therefore use the Amazon Mechanical Turk to obtain inexpensive human taggers. Although the taggers are not experts and the tagging process is not carefully controlled, we believe that utterances with high inter-annotator agreement can be assumed to be reliably tagged.

The initial effort described in this paper is relatively small and exploratory – only a total of 100 sentences were tagged. Nevertheless the results obtained are informative. All sentences and the obtained tags are being made public. It is our intention to follow this up with a much larger annotation task where a corpus of several tens of thousands of sentences will be tagged. These sentences too will be made freely available to public.

The rest of the paper is arranged as follows. In Section 2 we describe how we generate the dataset. In Section 3 we describe our setup for obtaining annotations, including the queries asked. In Section 4 we analyze the obtained results. Finally in Section 5 we provide our conclusions.

2. A LINGUISTIC-PLAUSIBILITY DATASET

Linguistic plausibility as we consider it in this paper is subjective. Therefore, we will not attempt to define it formally. Rather, linguistic plausibility is approximately language for which people would often answer “yes,” to the question of: Is this something a person might say? The basic units of this dataset are “sentences.” The concept of sentence connotes grammaticality and at least a modicum of semantic coherence, so perhaps “utterance” or “word sequence” better describes the units of this dataset. However, we will refer to them as “sentences” in this paper.

Since written human language typically has some degree of meaning and coherence, we cannot simply harvest sentences from human-written or human-spoken language. It will not provide the desired negative examples. Instead, we artificially generate random sentences using a statistical N-gram LM. This ensures that the sentences, on the surface, appear to be real human-written sentences, and sometimes do make sense (*i.e.* are clearly plausible). However, the lack of a real, human author ensures that the data contains plenty of “nonsense” sentences, as well as many with intermediate degrees of sense.

We use a 3-gram language model derived from Web N-gram counts provided by Google through LDC [7]. The motivation behind using Google N-grams is that they have been derived from massive amounts of text and are possibly the best representation of domain-independent relative frequencies of word N-grams currently available. We can therefore expect to generate text that is not specific to any domain and may even switch topics within a sentence – a phenomenon that is not infrequent in human speech.

Here are a few examples of sentences randomly generated by our model:

- *All departments babies*
- *Back to the islands as well as our new app server*
- *Driven by the listers or their representatives are so many can say yes*

Very short word sequences generated by this model are typically difficult to make sense of one way or another, and long word sequences are almost always nonsense since the LM has more opportunities to jump from topic to topic. Because of this, we limit the dataset to randomly generated sentences of moderate length: eight to twelve words long. Additionally, we conflated differences in case, and removed tokens containing non-alphabetic characters. Here is one sentence generated by this model that was overwhelmingly judged to make sense:

- *From casual to classic in design and preparation*

Here is one that was overwhelmingly judged to be nonsense:

- *A small fishing village in the same day delivery gift shop*

3. TAGGING THE DATASET WITH NON-EXPERT ANNOTATORS

The Amazon Mechanical Turk was employed to tag the generated sentences. The Mechanical Turk is a web service that acts as an exchange where subscribers may post tasks which will be performed by other subscribers, who we will call “providers”, for a fee. The Turk provides various mechanisms for determining the number of tasks performed by a particular provider, to ensure that tasks do not get posted to the same provider twice, to time tasks etc. In our task, automatically generated sentences were posted on the Mechanical Turk and providers were instructed to annotate their plausibility. While it is also possible to restrict the geographic location of providers and to employ other such filters on the Mechanical Turk, we did not filter the providers, ensuing only that the same annotation task was not sent to any person twice. As a result, there was no guarantee of linguistic expertise on the part of the annotators evaluating the sentences; most of them were almost certainly non expert.

Due to the inherent subjectivity of this task, one of the biggest challenges is how to elicit as consistent responses as possible from non-experts. Intuitively, plausible sentences must be sensible. In our experience, it is exceedingly confusing to non-experts to simply ask them “does this sentence make sense?” This is not surprising; in the appropriate context, almost any sequence of words generated might be considered to make sense, or not make sense. For instance, just as “flying a broom to London” might make sense in a Harry Potter novel, “their spouses listed in the Ministry of Health and Racquetball Courts” would make sense in a country that has a ministry of health and racquetball courts!

This subjectivity may be exacerbated when judges are asked to actually think about whether or not a sentence makes sense. That is, the more one thinks about such things, the foggier their conclusions are and the less certain they become about their decisions. Linguists who are familiar with doing grammaticality judgments are likely familiar with this phenomenon: the more one thinks about the grammaticality of a sentence, the harder it becomes to remember and decide what is grammatical and what is not.

So, on the one hand, we would like to “trick” people into answering using their “gut” instinct without thinking too hard, but on the other hand, the task needs to be concrete, clear, and directed enough that our judges are neither confused nor neglecting the task through carelessness. The problem thus becomes that of posing questions to the annotators in a manner that will elicit a consistent and meaningful response.

3.1. Methods of eliciting sensicality judgments from subjects

Given the above factors, we posted generated sentences with five different questions. Each annotator was presented with one of the questions and a set of sentences to tag. No annotator got the same sentence twice or two different questions. The exact text of each question can be found on the first author's website . The questions we asked are the following. Identifiers for each are listed in bold, for convenience.

- **Valid:** You will be presented with a series of word sequences. Please indicate if they are
  a nonsense sequences or
  b valid sentences or valid parts of longer sentences.
- **Overhear:** Is this something you might plausibly overhear in a conversation or read on the Web?
- **Turing:** Do you believe this sentence could have been generated by a person, or was this automatically generated by a computer program?
- **Scale:** Rank the sentence, on a scale from 1 to 5, where 1 is very semantically valid, and 5 is complete nonsense.
- **Describe:** Describe in your own words a situation in which this sentence makes sense.

The motivation behind the fifth question is slightly different than the others. In questions one through four, we ask the question directly, in several different ways. In the fifth, we ask subjects to describe a situation in which the sentence "makes sense." Our goal in asking this question is to time our subjects, and see if it takes them longer to "make sense" of more nonsensical sentences.

The timing in this fifth question is inspired by psychological experiments on human response to sensicality. Findings in the field of psychology show that people tend to take more time to process nonsense sentences in a self-paced reading task [8]. We found a slight correlation between response time and answers to questions one through four, but it was either not very pronounced, or our ability to measure it using Mechanical Turk was insufficient. The goal of asking people to describe the context was generally to extend the entire length of the task, with the hope that stretching out the duration would cause a stronger differential to emerge.

There are many factors in reading time that we do not directly control for. For instance, subjects are free to perform tasks as quickly or slowly as they wish. They might walk to the refrigerator to retrieve a beer in the middle of answering a questions. Sentence lengths are also not precisely controlled (only roughly, to 8 to 12 words).

4. RESULTS AND ANALYSIS

We randomly generated 100 sentences using the model described in section two. For each of the questions above, we created one batch of 100 HITs. We asked 10 people to perform each HIT, for $0.01 each. We found that at this rate, all 1,000 responses would be complete in approximately four or five hours. We note that this is only a preliminary data set; a much larger data set is planned based on the results of this study.

The table below shows the average response obtained with each of the five questions. For instance, using the question "valid", a total of 48.9% of responses tagged sentences as plausible. Similarly, for the question "scale", the average response value was 3.2.

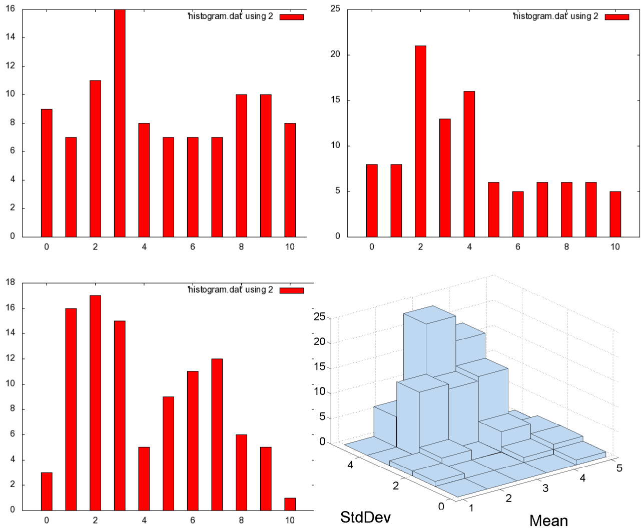

Fig. 1. The top row and the bottom left plots are response histograms for "valid", "overhear" and "turing". Bottom right: 3D histogram for responses for "scale".

Method	Avg. Time	% plausible
Valid	10.0	48.9
Overhear	11.9	40.9
Turing	10.0	43.2
Scale	11.7	3.2
Describe	53.9	-

More than the average time, we are interested in characterizing the inter-annotator agreement for each of the questions.

Figure 1 shows the histogram of responses for the "valid", "overhear", "turing" and "scale" queries. In the histograms for "valid", "overhear" and "turing" the x-axis represents the number of respondents who tagged a sentence as "plausible", and the y-axis value of the histogram at any x shows the number of sentences with the corresponding number of positive ("plausible") responses. Ideally, all judges would unanimously tag each sentence as either "plausible" or "implausible". In such a case, each of the histograms would have two peaks, one at 0 and the other at 10. A histogram with a peak to the center (at 5 and 6) indicates that annotators are generally unable to decide about the answer to the query and that the response is random. In all figures the histogram is relatively flat, showing that although the annotators were not unanimous in their tags, there were some sentences for which the responses were unanimous or close to unanimous. Overall, the histogram for the "valid" query appears to be least humped to the middle, indicating that it may have the highest inter-annotator agreement. This is borne out in the inter-annotaor agreement analysis we perform in the next subsection. In terms of unanimous responses, the "valid" and "overhear" questions both resulted in over 50% of sentences being tagged with 80% or greater unanimity between annotators.

The bottom right panel of Figure 1 shows a three-dimensional histogram for the query "scale". The x-axis is the mean response to queries, quantized to an integer between 1 and 5. The y-axis is the standard deviation of the responses to queries, also similarly quantized. The z-axis value at any (x, y) shows the number of queries for which the mean score and standard deviation were x and y respectively. If the inter-annotator agreement were high, the histogram would be largely biased towards the left of the figure. If sentences were largely scored as having high "plausibility" or "implausibility",

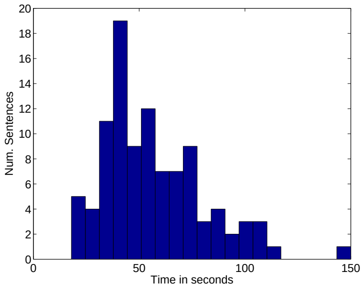

Fig. 2. Histogram of mean time to answer the "describe" question.

the histogram would be biased towards the upper and lower edges of the figure. Not surprisingly, we find that when given a sliding score, most annotators are uncertain (with a mean score of 3.2). Also on a numeric scale, there is greater disagreement between annotators.

Although the "describe" question elicits text responses, it is primarily intended as a means of identifying the "taggability" of sentences. The longer it takes for a person to respond, the harder it is for them to justify their response. Thus, if the mean response times is low for a sentence, it is probably clearly plausible (or implausible), but high mean response times indicate a sentence for which this judgement is potentially fuzzy. Figure 2 shows a histogram of the average response time for the sentences. Curiously, it shows that when asked to describe the reason for their tagging, most annotators took relatively short time indicating a fair degree of certainty in their tagging.

4.1. Inter-annotator agreement

To actually quantify the inter-annotator agreement for the three binary questions, we follow the methods described in Artstein and Poesio's (2008) survey [9] of inter-annotator agreement for computational linguistics. For this data collection effort, we have up to ten annotators per sentence, so we use the methods described for evaluating agreement among multiple annotators, in particular, multi-π and multi-κ , which are multiple-annotator versions of William Scott's π coefficient, and Jacob Cohen's κ statistic.

The raw agreement score is pairwise; that is agreement between each pair of annotators of a sentence. Thus, if nine annotators choose "valid" and one chooses "invalid," we get an overall pairwise agreement of 80%. The multi-π statistic assumes that annotators do not have their own bias, but allow the categories to be non-uniformly distributed. The multi-κ statistic allows each annotator to have his or her own biases. We slightly modified the multi-π metric ("agreement") because unlike in a conventional data annotation scenario, we do not have the same ten people annotating each sentence. For completeness, we report all three measures. Intuitively, the π and κ statistics represent proportionally how far the agreement is beyond chance agreement, on the way to perfect agreement. We do not compute inter-annotator agreement for scale and context.

Method	Agreement	Multi-π	Multi-κ
Valid	.664	.327	.329
Overhear	.636	.443	.250
Turing	.612	.4218	.212

The table above shows the inter-annotator agreements for the three questions. Both the kappa and pi metrics show a "fair" degree of agreement between annotators for all three questions. However, the "Valid" question rates the best among all three.

For the purpose of the proposed corpus, we are particularly interested in sentences in which inter-annotator agreement is high. The table below shows inter-annotator agreement for questions when only sentences for which a minimum of 8 annotators agreed were considered. Not surprisingly, the inter-annotator agreements are much higher, with values that are considered indicative of "substantial" agreement for both "Valid" and "Overhear".

Method	Agreement	Multi-π	Multi-κ
Valid	.846	.690	.693
Overhear	.785	.760	.556
Turing	.760	.382	.512

5. CONCLUSIONS

None of the questions are particularly superior to the others in eliciting clear binary response. In general "Valid", arguably the simplest of the three questions, appears to generate slightly most consistent responses. We note that sentences are often difficult to tag, as indicated by responses to "scale" and "describe". Given the difficultly that human subjects seem to have in identifying valid sentences, it leads one to wonder what the limitations of a computational model for language might be. On the other hand, the differences in response patterns to "scale" and "describe" also indicate that when annotators are asked to articulate the reasons for their tags, they tend to be more certain, suggesting that a deeper undelying principle may exist.

Nevertheless we believe the data set, if collected, will be an invaluable tool for NLP researchers, particularly those working in speech recognition and MT. The near-unanimously tagged subsets can be useful for discriminative training approaches, but the data with uncertain tags too can be valuable for the analysis of the confusions inherent in language. We expect to collect a much larger tagged data set of the kind discussed here. The current data (along with tags) are publicly downloadable from http://mlsp.cs.cmu.edu/projects/mechanicalturk.

6. REFERENCES

[1] E. Brill, R. Florian, J. C. Henderson and L. Mangu, "Beyond N-grams: Can linguistic sophistication improve language modelling?", COLING 1998.

[2] R. Rosenfeld. "Adaptive Statistical Language Modeling: A Maximum Entropy Approach," Ph.D dissertation, CMU, 1996

[3] C. Chelba and F. Jelinek. "Structured language modeling," Computer Speech and Language, 2000.

[4] H. Erdogan, R. Sarikaya, S. Chen, U. Gau and M. Picheny, "Using semantic analysis to improve speech recognition performance," Computer Speech and Language, Vol 19:3., July 2005

[5] H. T. Ng, D. C. Y. Lim and S. K. Foo. "A Case Study On Inter-Annotator Agreement For Word Sense Disambiguation,' SIGLEX Workshop On Standardizing Lexical Resources, 1999.

[6] D. Gildea and D. Jurafsky, "Automatic labeling of semantic roles," In Proceedings of ACL, 2000.

[7] T. Brants and A. Franz, "Web 1T 5-gram Version 1.," Linguistic Data Consortium, Philadelphia, 2006.

[8] B. McElree, M. J. Traxler, M. J. Pickering, R. E. Seely RE and R. Jack-endoff, "Reading time evidence for enriched composition," Cognition. 2001;78(1):B17-25.

[9] R. Artstein, and M. Poesio, "Inter-coder agreement for computational linguistics," Comput. Linguist. 34, 4 (Dec. 2008).