Modeling the Effects on Time-into-Utterance on Word Probabilities

Nigel G. Ward, Alejandro Vega

Department of Computer Science, University of Texas at El Paso

nigelward@acm.org, avega5@miners.utep.edu

Abstract

Most language models treat speech as simply sequences of words, ignoring the fact that words are also events in time. This paper reports an initial exploration of how word probabilities vary with time-into-utterance, and proposes a method for using this information to improve a language model. This is done by computing the ratio of the probability of the word at a specific time to its overall unigram probability, and using this ratio to adjust the n-gram probability. On casual dialogs from Switchboard this method gave a modest reduction in perplexity.

Index Terms: dialog, Switchboard corpus, time-based language model, perplexity

1. Introduction

Speaking is a cognitive process in time and also a communicative process in time [1], but these processes are not directly modeled by today's language models, which generally treat speech as consisting merely of sequences of words. However, cognitive and communicative considerations can affect when various words are likely to appear. For example, the fact that the production of speech involves mental effort leads us to expect fillers and semantically light words early in utterances, with words referring to complex concepts or thoughts involving memory retrieval or reasoning tending to appear later. The fact that speech is typically organized so as to be readily comprehensible by the listener also provides expectations, for example, grounding is likely to happen early in utterances [7], with complex content or disaffiliating phrases delayed to later.

Thus we expect certain words to be relatively more common early in utterances, and others more common later. If so, this may be useful for speech recognition, incorporated in a language model in combination with other information. Although the idea of conditioning word probabilities directly on time-into-utterance is novel as far as we know, this idea fits in with other attempts to improve on n-gram models by allowing wider contextual information, e.g. [2, 3, 4, 5, 6]. Also of note is Ma and Meteer's attempt to exploit a communicative principle, that given information generally precedes new information, for language modeling [7].

This paper reports an initial exploration of how word probabilities vary over time and how this can be used to improve a language model. Section 2 illustrates how word probabilities vary with time-into-utterance, Section 3 explains how time-based probabilities can be combined with n-gram probabilities, Section 4 presents and discusses the results, and Section 6 points out directions for future work.

2. Initial Observations

We used the ISIP transcriptions of the Switchboard corpus, a collection of short telephone conversations on light topics between mostly unacquainted adults [8, 9]. We split each track into utterances, arbitrarily defined as sequences of words delimited by at least 1 second of silence both before and after, using the regions labeled *[silence]* in the transcripts and merging adjacent silence regions. We then tagged each word by the time from the start of utterance to the start of that word: conceptually each utterance was split into buckets. For example, words which began between 0 and 0.1 seconds into the utterance were tagged as belonging to bucket 0, those beginning between 0.1 and 0.2 seconds as belonging to bucket 1, and so on.

We computed the bucket probability (time-based probability) $P_{tb}(w_i@t)$ for each word as its count in the bucket for t divided by the total in that bucket:

$$P_{tb}(w_i@t) = \frac{count(w_i@t)}{\sum_j count(w_j@t)} \quad (1)$$

Using the counts in 1000 Switchboard tracks, Table 1 shows that the most common words do indeed vary with time-into-utterance.

The difference between a time-based probability and the standard unigram probabilities can be conveniently expressed by their ratio:

$$R(w_i@t) = \frac{P_{tb}(w_i@t)}{P_{unigram}(w_i)} \quad (2)$$

Figure 1 illustrates how this ratio can vary over time.

These facts suggest that an improved probability estimate may be obtained for a hypothesized word at time i by using the bucket probability of that word at i instead of the general unigram probability. Initial experiments in our laboratory by Shreyas A. Karkhedkar showed that this was the case when sufficient data was available. In combination with back off to general unigrams in cases of insufficient data, time-based probabilities gave a reduction in perplexity on Switchboard data from 481.6 to 470.4.

3. Combination with Trigrams

Thus it seems that conditioning on time-into-utterance can provide useful information. The next question is whether this information is non-redundant to that captured by a more powerful model, trigrams. As our baseline model we used the default SRILM order 3 (trigram) backoff model [10].

Our first attempt to use time-based information combined it with the backoff model by interpolation, using a simple weighted average. However this performed poorly; as the trigram probability estimates were generally quite good, crudely averaging them with a weaker model was counterproductive.

Instead we decided to use the time-based probabilities merely to tweak the backoff probabilities. We use a scaling factor derived from R to determine how much to tweak. For example, for a word occurring at time t, if the bucket probability

0-.5	.5-1	1-1.5	1.5-2	2-2.5	4-4.5	8-8.5	16-16.5
yeah	I	I	I	and	and	and	and
I	the	the	the	the	I	I	the
and	a	and	and	I	the	the	I
you	to	a	to	to	a	to	you
uh	and	to	a	a	to	you	to

Table 1: The Five Most Frequent Words in Selected Buckets. Times are in seconds.

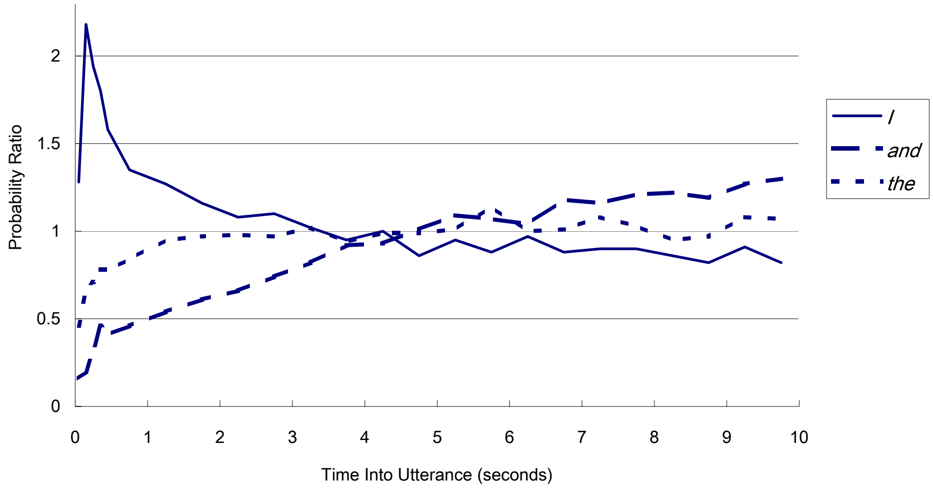

Figure 1: Frequency ratio R vs. Time-Into-Utterance for the Three Most Common Words. Words at utterance start (time-into-utterance = 0.0) are excluded. The rightmost points represent the range 9.5 seconds and up.

bucket	high-S words	low-S words
ϵ–0.1s	don't, that's, know, think, well, you, was, yeah, do, have . . .	with, out, or, on, be
0.1–0.2s	okay, yes, that's, sure, yeah, really, haven't, don't, think, can't . . .	over, money, every, care, anything
0.2–0.3s	okay, right, yes, yeah, no, see, that's, sure, i-, well . . .	minutes, [laughter-okay], home, everything, dear
0.3–0.4s	great, right, uh-huh, yes, well, okay, no, that's, haven't, yeah . . .	day, school, things, year, bit
0.4–0.5s	uh-huh, great, right, okay, well, yes, yeah, that's, no good . . .	come, stuff, her, every, day
0.5–1.0s	um-hum, uh-huh, agree, yeah, huh, yes, definitely, okay, heard, well . . .	program, jury, child, whether, weeks
1.0–1.5s	uh-huh, huh, bye-bye, bet, um-hum, yeah, exactly, isn't well, oh . . .	man, everybody involved education during
1.5–2.0s	bye-bye, huh, friends, talked, yes, problem, funny, age, tell . . .	basically, education, change, twelve, hand
2.0–2.5s	today, night, huh, though, mine, supposed, while, Texas, remember, i[t]- . . .	together, places, might, couldn't, moved
2.5-3.0s	Texas, times, program, huh, high, movie, insurance, system, enjoy, name . . .	feel, life, best, whatever, stay
3.0–3.5s	until, college, usually, basically, ago, try, gone, lived, made, fact . . .	isn't, person, percent, thinking, thirty
3.5– 4.0s	thirty, myself, huh, week, part, lived, last, state, spend, run . . .	um-hum, fun, thinking, great, enjoy
4.0–4.5s	call, month, took, usually, movie, called, child, Texas, ten, someone . . .	being, um-hum, own, goes, huh
4.5–5.0s	movie, since, system, started, life, working, might, point, doing, different . . .	great, um-hum, may, love, am
5.0–5.5s	couple, college, years, times, bit, whatever, money, year, both, Dallas . . .	okay, still, gets, away, idea
5.5–6.0s	ago, somebody, times, year, try, college, actually, least, i'll, being . . .	great, okay, may, interesting, love
6.0–6.5s	country, own, does, while, pay, need, everything, husband, went, stuff . . .	trying, started, great, anyway, yes
6.5–7.0s	few, look, house, care, away, why, watch, hundred, couple, enough . . .	four, didn't, sometimes, um-hum, started
7.0–7.6s	ago, week, has, always, being, whatever, try, times, six, wasn't . . .	area, oh, also, yes, uh-huh
7.5–8.0s	four, wasn't, usually, different, better, take, most, few, after, two . . .	yeah, um-hum, yes, another, uh-huh
8.0–8.5s	whatever, everything, having, through, being, come, stuff, first, either, need . . .	interesting, too, did, uh-huh, um-hum
8.5–9.0s	dollars, come, were, house, five, twenty, these, last, first, before . . .	okay, oh, live, interesting, um-hum
9.0–9.5s	his, hard, these, different, doesn't, sort, before, back, school, live . . .	right, yeah, okay, will, um-hum
9.5s-∞	authority, shirts, obvious, whereas, pants, corn, losing, bottle, percentage, match . . .	hi, minutes, [laughter-okay], dear

Table 2: Characteristic and Uncharacteristic Words in Various Buckets, that is, words with the highest and lowest S values.

word	start	bucket	R	S	$P_{backoff}$	P_{bs}	P_n	benefit
well	0.00	–	–	–	.056	.056	.056	—
I	0.15	0.1–0.2s	2.18	1.26	.201	.254	.221	+042
hadn’t	0.25	0.2–0.3s	1.08	1.00	.001	.001	.001	–.049
either	0.55	0.5–1.0s	2.65	1.34	.005	.007	.006	+.102
we	1.39	1.0–1.5s	0.98	1.00	.004	.004	.004	+.000
hadn’t	1.51	1.5–2.0s	0.29	1.00	.001	.001	.001	–.001
you	1.88	1.5–2.0s	0.95	0.99	.007	.007	.007	–.016
know	1.97	1.5–2.0s	0.82	0.94	.454	.427	.438	–.015
like	2.19	2.0–2.5s	1.11	1.03	.018	.018	.018	+.014
I	2.41	2.0–2.5s	1.08	1.02	.086	.088	.088	+.009
said	2.52	2.5–3.0s	1.27	1.06	.278	.295	.291	+.086
we	2.66	2.5–3.0s	0.93	0.98	.023	.023	.023	–.008

Table 3: Example of the Computation of P_n on a Fragment of an Utterance. The “benefit” is the log of the ratio of P_n to $P_{backoff}$.

indicates that the word is more common at t than at other times, then we multiply the backoff probability by a scaling factor to reflect this. This gives the “bucket-scaled” backoff probabilities:

$$P_{bs}(w_i@t|c) = S(w_i@t)P_{backoff}(w_i|c) \tag{3}$$

where c is the local context, specifically here the preceding two words, and S is the scaling factor, explained below.

The scaling factor is based on R, but we do not use R directly as the scaling factor, for two reasons. First, R is less informative in cases where the bucket probability is based on sparse counts, as for infrequent words or in those in late buckets. To estimate the informativeness we use the χ^2 test to evaluate the hypothesis that the number of occurrences of the word in the bucket differs from that expected from the bucket size and the unigram probability of the word. We compute the P-value of this hypothesis, p, and from that our confidence in the hypothesis: $q = 1 - p$. (If the expected count of the word in the bucket is less than 5, then we have no confidence, and we set q to 0.) We then raise R to the q^{th} power; thus, if the confidence in the bucket probability is low, then S will be close to 1 and the time-based information will have little effect.

The second complication in the computation of S is because the time-based estimate and the backoff estimate are not independent. We therefore raise R to a constant power k less than 1 to attenuate the impact of the bucket-based probability on the backoff probabilities. Empirically 0.3 is a good value for k, although the results are not that sensitive to this parameter.

Thus,

$$S(w_i@t) = R(w_i@t)^{kq} \tag{4}$$

One necessary detail is smoothing: if the count in some bucket for some word is 0 we replace it with 1. This ensures that R is never 0, which is required to make equation 4 well-behaved. No explicit discounting is done, since discounting happens as a side-effect of normalization. Table 2 shows words with extreme S values in each bucket.

Finally there is a normalization step to ensure that all the probabilities across the vocabulary add to 1 in each bucket. This is done at runtime: when looking up the probability for a word, the bucket-scaled backoff probabilities for all the words in the corpus are computed, and the bucket-scaled backoff probability of the word of interest is divided by the sum. This gives the normalized combined probability, P_n:

$$P_n(w_i@t|c) = \frac{P_{bs}(w_i@t|c)}{\sum_j P_{bs}(w_j@t|c)} \tag{5}$$

Since the values of P_{bs} depend on the preceding words as well as the time-into-utterance, they cannot be pre-computed: they must be calculated for each word in the vocabulary. This normalization phase, needed here for the sake of fair perplexity calculations, makes the amount of computation non-trivial; however this might not be needed in a speech recognizer.

Bucket-based scaling is not applied if a word occurs at the start of an utterance. The reason is that in this position the probability is accurately modeled by the bigram *<s> word*: the fact that the word is also in bucket 0 brings no new information. As time-based scaling thus has nothing to offer such words, they are not used for training either. Specifically they are not included in the bucket 0 counts nor in the unigram counts, thus they do not contribute to the computation of P_{tb} or anything else.

For test purposes we used a model with 24 buckets: 5, each 0.1 seconds in width, from 0 to 0.5; 18, each 0.5 seconds in width, from 0.5 to 9.5; and one from 9.5 seconds out to infinity.

For the experiments computation time was an issue, so the vocabulary was limited to 5000 words; other words were treated as unknown and excluded from all computations. The time-based adjustments were implemented as a wrapper around the function NgramLM::wordprobBO in the SRILM toolkit [10]. Table 3 illustrates how these computations work.

4. Results

The test set was 24 tracks from Switchboard, representing about 115 minutes of speech and including 10174 words. For evaluation purposes we ignored out-of-vocabulary words and sentence-end tags. As seen in Table 4 the perplexity was lower for the normalized combined model, indicating that the time-based probabilities are improving the model. Overall the bucket-based scaling benefited the estimates for 5124 tokens and hurt 3649.

Pending systematic analysis, we scanned through the effects of time-based scaling and found some recurring patterns.

In general, utterances which seemed to be typical of the casual small talk genre dominating Switchboard were often scored higher, and those less typical were often scored lower. For ex-

	perplexity
Standard, $P_{backoff}$	127.833
Time-Based, P_n	127.495

Table 4: Evaluation Results

ample, the estimates were hurt for every word in the fluent, grammatical and swift utterance *he does that every year*, especially for the word *every* occurring at 0.48 seconds in, since in Switchboard *every* more typically occurs late in utterances.

Sometimes there are words which appear to start a new utterance, in some sense, but which are not preceded by a second of silence. These include words that seem to occur more as a response to something said by the interlocutor than as a result of the progress of the speaker's own cognitive and production processes. For example, in . . . *sometimes ten to fifteen percent of the an[d]- yeah and and you know the one of the things I remember* . . . , the word *yeah* was not preceded by a second of silence. In such cases the bucket-scaling typically decreases the probability of the "initial" word, here *yeah*, decreasing performance. A more sophisticated definition of utterance start, using information about the behavior of the interlocutor, may be able to overcome this problem. Another possibility is to use prosodic information, for example whether the previous word was drawn out, fading off, or in a low flat pitch.

Sometimes there are words which are much better modeled by trigrams, the word *know* in Table 3 being an example. Here the scaling factor decreases the probability because *know* is uncommon around 2 seconds in. What the model is failing to capture is that the bigram *you know* is in fact common around this time. It may be possible to alleviate such problems by basing the scaling factor not only on the bucket-based unigrams but also on bucket-based bigrams, although the sparseness problem would limit this to the most frequent bigrams.

5. Summary and Future Work

Observing that word probabilities depend on time-into-utterance, we proposed a way to use this information to improve a standard trigram model. The resulting improvement in perplexity shows that time-into-utterance brings information which a standard trigram model does not capture.

Our model clearly could be refined. In many cases the way that probability varies with time-into-utterance seems to be similar across a number of words, suggesting the use of a class-based model. A small study by Nisha Kiran in our laboratory found that words in the first half second of utterances have significantly higher affect [11] on average than words occurring later, suggesting a model using the semantic or pragmatic component dimensions of words. We could also optimize various details of the model, such as the widths of the various buckets, the length of the pause used in the definition of utterance, and the computation of the scaling factor. It may also be worthwhile to build an adaptive model, adapting to speaking rate, to speakers or to genres.

The techniques developed here may be useful for time-based language modeling more generally. Beyond time-into-utterance, other features may be useful, such as time since the most recent end of an interlocutor's turn, time since disfluency recovery, time since last content word, time since various prosodic features, time-into-dialog, or time until end of utterance.

We hope that this model, or other time-based language models, will have practical utility for, e.g. improving speech recognizer performance, supporting the generation and synthesis of more natural and comprehensible utterances, and improving speaker recognition.

6. Acknowledgments

We thank Shreyas Karkhedkar for the initial implementation, Nisha Kiran for the affective analysis, Olac Fuentes and two Interspeech reviewers for comments, and Joe Picone, Andreas Stolcke, and Margaret Bradley and Peter Lang for making available the Switchboard labeling, the SRILM toolkit, and the ANEW data, respectively. This work was supported in part by an REU Supplement to NSF Grant IIS-0415150.

7. References

[1] Clark, Herbert H., Speaking in Time. *Speech Communication*, 36, pp 5–13, 2002.

[2] Gildea, Daniel & Hofmann, Thomas. Topic-Based Language Models Using EM. Eurospeech 1999.

[3] Schwenk, Holger & Gauvain, Jean-Luc. Neural Network Models for Conversational Speech Recognition. in Proc. Interspeech 2004.

[4] Ji, Gang & Bilmes, Jeff. Multi-Speaker Language Modeling. HLT 2004.

[5] Ji, Gang & Bilmes, Jeff. Backoff Model Training using Partially Observed Data: Application to Dialog Act Tagging. HLT/NAACL 2006.

[6] Singh-Miller, Natasha & Collins, Michael. Trigger-Based Language Modeling Using a Loss-Sensitive Perceptron Algorithm. IEEE ICASSP 2007.

[7] Ma, Kristine W., Zavaliagkos, George & Meteer, Marie. Bi-Modal Sentence Structure for Language Modeling. *Speech Communication*, 31, pp 51-67, 2000.

[8] Manually Corrected Switchboard Word Alignments. January 29, 2003. ISIP, Mississippi State University. retrieved from http://www. ece.msstate.edu/research/isip/projects/ switchboard/

[9] Godfrey, J. J., Holliman, E. C., & McDaniel J. Switchboard: Telephone speech corpus for research and development. Proceedings of ICASSP, pp. 517-520, 1992.

[10] Stolcke, Andreas. SRILM - An Extensible Language Modeling Toolkit, in Proc. Intl. Conf. Spoken Language Processing, 2002.

[11] Bradley, Margaret M. and Lang, Peter J. Affective Norms for English Words (ANEW): Instruction Manual and Affective Ratings. Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, 1999.