Grounding language models in spatiotemporal context

Brandon C. Roy, Soroush Vosoughi, Deb Roy

The Media Lab, Massachusetts Institute of Technology
Cambridge, MA 02139

bcroy@media.mit.edu, soroush@media.mit.edu, dkroy@media.mit.edu

Abstract

Natural language is rich and varied, but also highly structured. The rules of grammar are a primary source of linguistic regularity, but there are many other factors that govern patterns of language use. Language models attempt to capture linguistic regularities, typically by modeling the statistics of word use, thereby folding in some aspects of grammar and style. Spoken language is an important and interesting subset of natural language that is temporally and spatially grounded. While time and space may directly contribute to a speaker's choice of words, they may also serve as indicators for communicative intent or other contextual and situational factors.

To investigate the value of spatial and temporal information, we build a series of language models using a large, naturalistic corpus of spatially and temporally coded speech collected from a home environment. We incorporate this extralinguistic information by building spatiotemporal word classifiers that are mixed with traditional unigram and bigram models. Our evaluation shows that both perplexity and word error rate can be significantly improved by incorporating this information in a simple framework. The underlying principles of this work could be applied in a wide range of scenarios in which temporal or spatial information is available.

Index Terms: language modeling, multimodal, spatial, temporal, context

1. Introduction

Language models are an integral part of many language processing applications such as speech recognition, information retrieval and machine translation. These models attempt to capture linguistic regularities inherent in natural language. However, natural language, especially spoken language, does not occur in a vacuum. As with almost all human activities, spoken language is situated in time and space and engages multiple modes of the human cognitive system.

Experiments in cognitive science and psycholinguistics have shown that various aspects of spoken language, such as the speaker's choice of words and syntax, are influenced by extralinguistic context. For example, previous research has shown that speakers tend to fixate on objects in a visual scene before they are mentioned [1]. More recently, Coco and Keller [2] have shown that the visual scan pattern of a speaker is predictive of what they will say.

There have been several attempts to combine these extralinguistic factors with traditional language and speech processing systems. Vosoughi [3] and Prasov et. al. [4] respectively incorporate head-pose and eye-gaze into various stages of speech recognition systems. These attempts have tried to model and replicate the process through which visual and linguistic systems are integrated. While one can try to model the multi-modal processes involved in language production, this is not an easy task since most of these processes are not accessible to standard research methods. However, these multi-modal interactions shape and structure the output of the language production system. These structures can be understood and modeled through statistical analysis of the speaker's words.

In this paper, we look at how space and time shape the output of the language production system. We then incorporate this information into standard language models to better predict a speaker's choice of words. This area of research is largely understudied although we feel it holds much potential. One reason there may be less work in this area is that a special type of corpus is needed for this kind of analysis. The corpus must be collected over a long period of time in order to capture temporal patterns of language use. For example, the likelihood of the word "coffee" is much greater in the morning than late at night. However this can only be observed consistently if there are enough instances of spoken language being used at different times during the day. Additionally, the corpus needs to be collected from a limited spatial domain so that the structures imposed by the spatial features become clearer. For the word "coffee", when observed in the domain of a household over an extended period, it becomes apparent that the word is more likely to be used in the kitchen than in the bedroom. Our corpus, which we describe in the next section, satisfies these conditions.

2. The Speechome Corpus

The Speechome Corpus is the corpus of transcribed speech collected for the Human Speechome Project (HSP) [5], a large-scale, naturalistic, longitudinal study of early language development. A custom audio-video recording system was installed in the home of a family with a newborn; recording started at the child's birth and continued for three years from 11 ceiling mounted cameras and 14 boundary layer microphones distributed throughout the house. Camera sensors had high dynamic range and captured roughly 15 frames per second and 1 megapixel resolution, although our video analysis used a subsampled 120x120 pixel format. Audio was digitized at 48 KHz at 16 bit resolution. Participants had full control of the recording system and privacy safeguards, generally turning the system on early in the morning and off just before bedtime. The resultant raw data spans roughly 1000 days and consists of more than 200,000 hours of combined audio and video recordings. However, most of our analyses focus on a subset of the data spanning the child's 9–24 month age range, a period of 488 days. We transcribed the speech during this period using BlitzScribe [6], a semi-automatic tool we developed to accurately transcribe data at scale. During this period, an average of 10 hours per day were recorded. The final Speechome Corpus consists of roughly 10 million words of transcribed speech from roughly 2

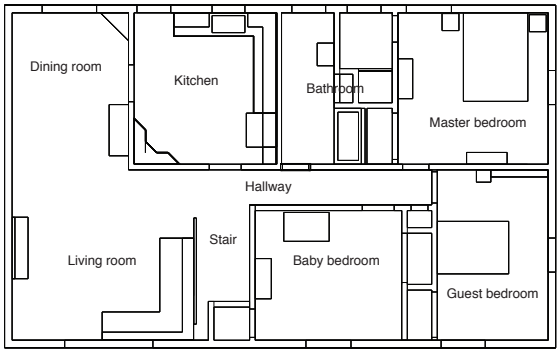

Figure 1: Floor plan of the primary floor where most activity took place in the Speechome Corpus.

million utterances, with most speech produced by the mother, father, nanny and child, although other speakers (e.g. grandparents) are also present in the data. Start and end timestamps and the source microphone ID were also stored with each utterance. The floor plan of the main living area is shown in figure 1.

For this work, we did not use video data directly but instead a processed version that grouped camera pixels into regions and indicated the presence or absence of motion in each region. To obtain this representation, background subtraction was first performed on each video frame to identify "active" pixels (i.e. pixels that had changed from their average value). The resultant stream of binary video frames from each camera was then processed to identify regions of correlated pixels, for a total of 487 regions across 9 of the 11 cameras in the home (omitting the master bedroom and bathroom). For each utterance in the corpus, a region activity vector was computed by accumulating and thresholding the number of active pixels in each region across all video frames within ± 5 seconds of the utterance start timestamp. The original development and details of this video processing pipeline can be found in [7].

With timestamps and region activity vectors for each of the more than 2 million utterances of spontaneous, natural speech, the Speechome Corpus is a unique resource for exploring spatiotemporal context in natural language.

3. Language models

A speaker or writer's choice of words depends on many factors, including the rules of grammar, message content and stylistic considerations. Most statistical language models do not attempt to explicitly model the complete language generation process, but rather seek a compact model that adequately explains the observed linguistic data. Probabilistic models of language assign probabilities to word sequences $w_1 \ldots w_\ell$, and as such the likelihood of a corpus can be used to fit model parameters as well as characterize model performance.

N-gram language modeling [8, 9, 10] is an effective technique that treats words as samples drawn from a distribution conditioned on other words, usually the immediately preceding $n-1$ words in order to capture strong local word dependencies. The probability of a sequence of ℓ words, written compactly as w_1^ℓ is $\Pr(w_1^\ell)$ and can be factored exactly as

$$\Pr(w_1^\ell) = \Pr(w_1) \prod_{i=2}^{\ell} \Pr(w_i | w_1^{i-1})$$

However, parameter estimation in this full model is intractable, as the number of possible word combinations grows exponentially with sequence length. N-gram models address this with the approximation $\tilde{\Pr}(w_i | w_{i-n+1}^{i-1}) \approx \Pr(w_i | w_1^{i-1})$ using only the preceding $n-1$ words for context. A bigram model ($n = 2$) uses the preceding word for context, while a unigram model ($n = 1$) does not use any context.

Even with this simplification, there may not be enough samples of rare words to obtain good probability estimates. One way to address this is to exclude rare words and replace them with a special symbol. Another technique is to apply a smoothing function that redistributes probability mass toward the lower frequency words. A third technique is "backoff" smoothing [11], in which an n-gram model falls back on an $(n-1)$-gram model for words that were unobserved in the n-gram context.

Since the model is probabilistic, model fit can be quantified using the corpus likelihood. A more commonly used measure that is related to likelihood is *perplexity*, defined as $PP = 2^{H(p,m)}$, where $H(p,m)$ is the cross-entropy [12] between the true data distribution p and the model distribution m; models with smaller perplexity better fit the data. Practically speaking, a good language model does a better job of predicting the next word in a sequence of words, a direct consequence of its better exploiting structure in the data.

3.1. Spatiotemporal context in language models

Spoken language, by its nature, occurs at a time and place, and both of these factors may play a role in language production. N-gram models capture some aspects of linguistic structure, but do not directly capture extralinguistic variables that may also be relevant. To investigate the value of spatiotemporal context in language modeling, we built a series of language models that incorporate space and time.

To begin, we built purely linguistic unigram and bigram models in Python, utilizing some components from NLTK [13]. These models used a vocabulary that was filtered to remove words occurring 5 or fewer times. Probability distributions were calculated using Witten-Bell smoothing [9]. Rather than assigning word w_i the maximum likelihood probability estimate $p_i = \frac{c_i}{N}$, where c_i is the number of observations of word w_i and N is the total number of observed tokens, Witten-Bell smoothing discounts the probability of observed words to $p_i^* = \frac{c_i}{N+T}$ where T is the total number of observed word types. The remaining Z words in the vocabulary that are unobserved (i.e. where $c_i = 0$) are given $p_i^* = \frac{T}{Z(N+T)}$. Although more advanced techniques such as Kneser-Ney and Good-Turing smoothing would likely yield better overall n-gram models [14], we chose Witten-Bell smoothing for its simplicity since our study is focused on incorporating spatiotemporal information and not optimizing n-gram model performance. In addition to Witten-Bell smoothing, bigram models also used backoff smoothing.

To introduce spatiotemporal context, we considered two approaches. The first was to simply treat discrete contextual variables as another condition in an n-gram formulation. For example, if the room were used as the spatial context and the preceding word as the linguistic context, they could be treated together as the conditioning context. However, this approach is less flexible, and can be problematic when the conditioning variable is not discrete. Another issue with such a model is the backoff procedure; if there is insufficient evidence for a spatial bigram, one backoff path is to a spatial unigram, another is to a non-spatial bigram. This issue is explored in [15].

For these reasons, we took a different approach to capturing nonlinguistic context. Using the Python Scikit-learn toolkit

[16], we trained a decision tree classifier [17, 18] with non-linguistic contextual features as input and the words as output classes. The basic intuition is that a good language model exploits the mutual information (i.e. dependency) between contextual variables and words, just as a good classifier exploits the mutual information between input features and output classes. Without knowing the best partitioning of an input feature space in advance, a classifier training algorithm might be able to find a good mapping from contextual features to words.

Decision trees represent a set of tests on an input feature vector that lead to a target classification. Each node in the tree is a test of a single feature, with the outcome selecting the branch to descend for the next test. The final classification occurs at the leaf nodes, where each leaf node corresponds to a target class. Although decision tree classifiers are discriminative, they can also provide probabilities over output classes. Decision trees are trained by finding the feature and feature value that best split the training data at each level in the tree. A good split is one that partitions the data into sets with minimal overlap between class labels. In our case, the decision tree finds a spatiotemporal partitioning of the training data where each partition has a low entropy word distribution. A good decision tree classifier will use an utterance's spatiotemporal context to predict its candidate words. For example, if an utterance occurred in the kitchen in the morning, the decision tree would select for breakfast related words such as "coffee".

3.2. Choosing spatial and temporal features

We experimented with several spatial and temporal features directly in Scikit-learn before using them in our language model. By using each word as a target class, and with thousands of words in the model vocabulary, we expected performance to be very low using standard classification error rates. Therefore, it was particularly important to evaluate classifier accuracy against baseline models to assess relative performance gains with different feature combinations. We constructed two baseline classifiers that we called RANDOM-CLS, which selected classes at random, and MODE-CLS, which always chose the most likely class from the prior class distribution. MODE-CLS corresponds to a classifier version of the unigram language model, in that it does not consider any context and simply chooses the most likely class.

For our purposes, the contextual classifier must outperform RANDOM-CLS to be useful, and should have comparable performance to MODE-CLS on test data. This is because outperforming RANDOM-CLS implies that the contextual classifier has extracted useful information from the nonlinguistic features, which may complement (but not necessarily surpass) the information provided by the unigram (MODE-CLS) model.

We experimented with different contextual features by comparing classifier performance to baseline classifiers. In the Speechome Corpus, each utterance has a start and end timestamp, but since utterances are all less than 10 seconds in length we used only the start timestamp to extract the following temporal features: year, month, day of month, day of week and time, represented as a decimal ranging from 0 to 24. We chose these features to capture fine to coarse level temporal patterns of daily life at home. For example, breakfast is a recurring pattern linked to a time of day, while paying bills is more closely tied to the day of the month. This combination substantially outperformed MODE-CLS in accuracy on a test set. Each utterance in the corpus also provides a list of active spatial regions, usually a small subset the full list of 487 regions. Although fine-grained

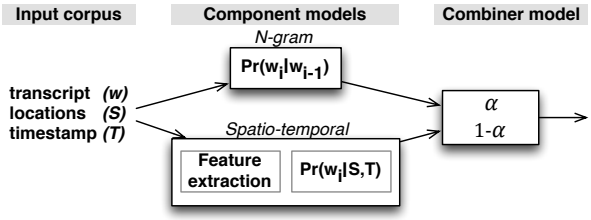

Figure 2: Model diagram showing the input data, the two component language models, and the combiner model which is parameterized by α.

spatial resolution may be useful in future work, we begin here by using the active regions to identify the room where activity is taking place. We chose rooms as our spatial feature since many household activities are linked to a particular room. For example, cooking takes place in the kitchen while entertaining guests takes place in the living room. The resultant spatial classifier outperforms the RANDOM-CLS and has comparable performance to the MODE-CLS.

3.3. Combined model

To obtain the final probability of a word in its spatial, temporal and linguistic context we built a combined model that simply mixed the probability distributions from the n-gram language model and the decision tree contextual language model using a mixing parameter $0 < \alpha < 1$. The output conditional distribution is thus

$$\Pr(w_i|w_{i-n+1}^{i-1}, S, T) = \alpha \Pr(w_i|w_{i-n+1}^{i-1}) \; + \; (1-\alpha) \Pr(w_i|S,T)$$

Our simple formulation could be extended to support different mixing parameters under different conditions; there are also other approaches to interpolating language models [19, 20] which we leave for future work. Figure 2 depicts the model architecture.

Varying α between 0 and 1 interpolates between the contextual model and the n-gram model. Choosing α to minimize perplexity on a held-out test set is one way to find a good α. Here, we do not automatically choose α since our main focus is on characterizing the contribution of extralinguistic information to overall model performance. In the following section we report perplexity as a function of the α parameter.

4. Results and discussion

We trained and evaluated a total of 8 different models. We started with unigram and bigram models, and then combined these n-gram models with the spatial, temporal and spatiotemporal models described above. These models were trained on 640,000 utterances randomly sampled from our corpus. The training set had a total of 5,847 word types and 1,828,892 total word tokens. The models were then evaluated on a held-out test set of 212,000 utterances (606,468 tokens). Figure 3 shows the performance of the combined models for different values of the mixture parameter α. These values are used to pick the best α for each combined model. Table 1 shows how well each model fits the test data (and the α used for each model). Models with smaller perplexity better fit the data.

Since the vocabulary used for each model is the same, the perplexity values can be compared between models. Through

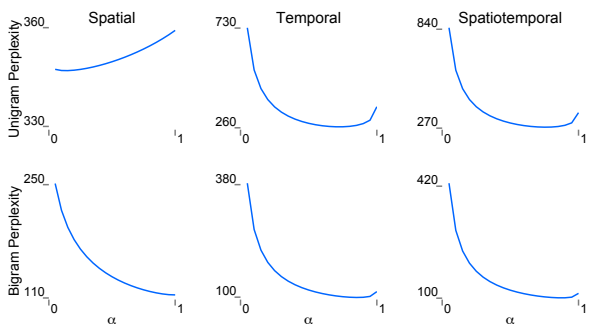

Figure 3: Perplexity scores as a function of the mixture parameter α for each combined model.

	Perplexity (using best α)	
Context	*Unigram*	*Bigram*
—	359.17	113.79
Spatial	346.95 ($\alpha = .15$)	114.07 ($\alpha = .95$)
Temporal	264.78 ($\alpha = .70$)	**99.26** ($\alpha = .85$)
Spatiotemporal	275.64 ($\alpha = .75$)	101.10 ($\alpha = .85$)

Table 1: Perplexity of unigram and bigram models when mixed with spatial and temporal features.

this comparison we can see that the introduction of spatial, temporal and spatiotemporal features into the unigram model improved model performance by 3.4%, 26.3% and 23.3% respectively. The performance of the bigram model was also improved by the incorporation of the temporal and spatiotemporal features (12.8% and 11.2% respectively), with the spatial feature having no significant effect.

In order to better understand the contributions of our contextual features, we took a closer look at the data. We considered word types that occurred at least 100 times in our dataset, which make up 546,382 of the 606,468 test word tokens (90%). For each of these word types we measured their average log probability under the various models (by averaging over all their tokens in the test data). Note that for log probability, larger values correspond to better model fit. We then calculated the difference between the word's average log probability under the unigram model and each of the combined models. These values indicate how much each word was helped or hurt by the addition of spatial, temporal and spatiotemporal information. Figure 4 shows the change in each word's average log probability, with positive values indicating improvement and negative values indicating a reduction in performance relative to the unigram model. As the figures show, most of the words are helped by contextual features.

We next took a closer look at some of the words to better understand these models. Table 2 shows the top ten most improved words for our three contextual features. The word "tea" is the most improved under the temporal model, perhaps because tea is most likely enjoyed at specific times in a household setting (one is reminded of the phrase "tea time"). Similarly, the word "diaper" is one of the most improved words under spatial (but not temporal) model. This is reasonable since the majority of diaper changing took place in the baby's bedroom (see figure 1 for the floor plan) but happened at various times during the day. Many of the other words correspond to playtime activities.

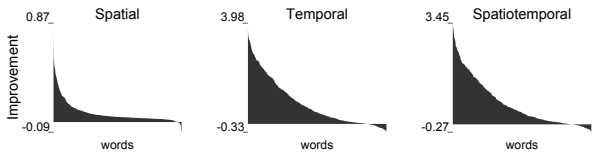

Figure 4: The change in word log probability between the contextual models and unigram model for each word occurring at least 100 times, in descending order of improvement.

Top Contextual Words (avg logprob improvement)

Spatial	*Temporal*	*Spatiotemporal*
pant (0.87)	tea (3.98)	bambi (3.45)
diaper (0.81)	bambi (3.80)	engine (3.30)
catch (0.57)	meow (3.56)	tea (3.30)
barney (0.54)	engine (3.43)	tractor (3.20)
mix (0.51)	tractor (3.42)	meow (3.11)
pee (0.50)	dump (3.35)	bump (3.07)
shower (0.48)	turtle (3.33)	dump (2.97)
rice (0.48)	quack (3.31)	spider (2.95)
spoon (0.46)	shake (3.31)	sock (2.93)
sugar (0.45)	farm (3.30)	turtle (2.89)

Table 2: Top 10 most improved words for each contextual model. Only words occurring at least 100 times in the corpus were considered.

5. Conclusions and future work

We have shown that incorporating spatiotemporal context into language models can yield substantially improved performance over standard unigram and bigram models. While the contributions of the spatial features were minor, especially when compared to those of the temporal features, we believe that the full potential of spatial information is not being realized in our current models. A possible extension of the model could include more sophisticated spatial feature selection techniques. Also, more advanced strategies can be used to mix the spatial and temporal classifiers with n-gram language models. One possible extension is to use separate α values for each word in the vocabulary, since the dependence of spoken language on spatial and temporal factors varies from word to word. While we saw that "tea" and "diaper" are strongly linked to temporal and spatial factors, other words such as "the" have almost no interaction with either factor. Incorporating this information may improve the overall performance of the spatiotemporal language model.

Although these models were trained and tested on a very unique corpus, the basic principles could be applied to any corpus where there is spatial or temporal information. In a broader sense, this paper illustrates that extending beyond linguistic features to incorporate other sources of context can improve the performance of language models.

6. Acknowledgements

The authors are grateful to George Shaw and Matt Miller for discussions and their work in processing the Speechome video corpus. We also wish to thank the Speechome transcription team and the anonymous reviewers for helpful comments.

7. References

[1] Z. M. Griffin and K. Bock, "What the eyes say about speaking," *Psychological science*, vol. 11, no. 4, pp. 274–279, 2000.

[2] M. I. Coco and F. Keller, "Scan patterns predict sentence production in the cross-modal processing of visual scenes," *Cognitive Science*, 2012.

[3] S. Vosoughi, "Improving automatic speech recognition through head pose driven visual grounding," in *Proceedings of the 32nd annual ACM conference on Human Factors in Computing Systems*. ACM, 2014, pp. 3235–3238.

[4] Z. Prasov and J. Y. Chai, "Fusing eye gaze with speech recognition hypotheses to resolve exophoric references in situated dialogue," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2010, pp. 471–481.

[5] D. Roy, R. Patel, P. DeCamp, R. Kubat, M. Fleischman, B. Roy, N. Mavridis, S. Tellex, A. Salata, J. Guinness, M. Levit, and P. Gorniak, "The Human Speechome Project," in *Proceedings of the 28th Annual Cognitive Science Conference*. Mahwah, NJ: Lawrence Earlbaum, 2006, pp. 2059–2064.

[6] B. C. Roy and D. Roy, "Fast transcription of unstructured audio recordings," in *Proceedings of Interspeech*, Brighton, England, 2009.

[7] M. Miller, "Semantic spaces: Behavior, language and word learning in the Human Speechome Corpus," Master's thesis, Massachusetts Institute of Technology, 2011.

[8] E. Charniak, *Statistical language learning*. MIT press, 1996.

[9] D. Jurafsky and J. H. Martin, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*, 2nd ed. Prentice Hall, 2008.

[10] C. D. Manning and H. Schütze, *Foundations of statistical natural language processing*. MIT press, 1999.

[11] S. Katz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 35, no. 3, pp. 400–401, 1987.

[12] T. Cover and J. Thomas, *Elements of information theory*. John Wiley & Sons, 2006.

[13] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python*. O'Reilly Media, Inc., 2009.

[14] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359 – 393, 1999.

[15] J. A. Bilmes and K. Kirchhoff, "Factored language models and generalized parallel backoff," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*. Association for Computational Linguistics, 2003, pp. 4–6.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[17] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[18] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, and R. Tibshirani, *The elements of statistical learning*. Springer, 2009, vol. 2, no. 1.

[19] R. Rosenfeld, "A maximum entropy approach to adaptive statistical language modelling," *Computer Speech & Language*, vol. 10, no. 3, pp. 187 – 228, 1996.

[20] B.-J. Hsu, "Generalized linear interpolation of language models," in *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*. IEEE, 2007, pp. 136–140.