Evaluating the role of non-lexical markers in GPT-2's language modeling behavior

Roberta Rocca
Aarhus University
University of Texas at Austin
roberta.rocca@cas.au.dk

Alejandro de la Vega
University of Texas at Austin
delavega@utexas.edu

Abstract

Transformer-based language models are often trained on structured text where non-lexical markers of sentence and discourse structure (e.g., punctuation and casing) are present and used consistently. Transformers encode these markers and arguably benefit from the information they convey. Yet, a systematic evaluations of the contribution of non-lexical markers to model performance, and of whether models' behavior changes significantly in their absence, is currently lacking. This knowledge is both relevant from a theoretical standpoint, but also important to understand how well pretrained models may perform in common application scenarios where casing and punctuation are absent or inconsistent. Here, we analyze GPT-2's language modeling behavior in parallel corpora that differ in the presence vs. absence of consistent punctuation and casing. We compute GPT-2's precision and uncertainty in next-token prediction for multiple context sizes, and compare the resulting performance distributions across corpora. We find that absence of non-lexical markers, especially punctuation, increases model uncertainty, and it affects (but does not catastrophically disrupt) GPT-2's precision in next-token prediction. Interestingly, the absence of non-lexical markers prevents the model from benefiting from larger contexts in order to reduce the uncertainty of its predictions. Future work will extend this paradigm to a wider range of models and systematically investigate how features of training text affect both language modeling and downstream predictive performance.

1 Introduction

The advent of Transformer-based language models (Vaswani et al., 2017) and their availability through high-quality easy-to-use libraries such as huggingface's transformers (Wolf et al., 2020) has widely democratized the use of state-of-the-art models beyond the NLP community. Transformers' language modeling capabilities can be leveraged off-the-shelf — with no further training and only minimal programming required — for a large variety of applications, ranging from neuroscientific investigations of human language processing (Merkx and Frank, 2020; Schrimpf et al., 2021) to interactive and improvisational storytelling (Austin, 2019).

Transformers (Devlin et al., 2019; Radford et al., 2019; Brown et al., 2020) are often trained on large corpora including highly structured text (e.g., BooksCorpus, (Zhu et al., 2015), or the English Wikipedia), where non-lexical sentence structure and discourse markers (punctuation and casing) are present and used consistently. Tokenization preserves these markers: punctuation is encoded through dedicated tokens and (for some models) casing is preserved through case-sensitive vocabularies.

Punctuation and casing encode rich information about sentence boundaries, internal sentence structure, and discourse (Steinhauer, 2003), which transformers' language modeling capabilities arguably benefit from. Yet, systematic investigations of whether this is the case, and how sparse or inconsistent use of these markers affects models' predictive performance, is lacking[1].

This knowledge would not only be informative from a theoretical standpoint (clarifying the contribution of non-lexical structure and discourse markers to transformers' language modeling capabilities) but also to understand whether popular pretrained models' capabilities generalize to common real-world application scenarios where non-lexical markers are absent or used inconsistently (e.g., social media text, or speech-to-text transcription). Discrepancies in performance could in fact be addressed by fine-tuning models on unstructured

[1]With the exception of studies on punctuation restoration (Courtland et al., 2020; Vāravs and Salimbajevs, 2018) and dialogue act recognition (Żelasko et al., 2021).

Proceedings of the 3rd Workshop on Evaluation and Comparison of NLP Systems, pages 96 - 102

baseline	no punctuation
the date: September eighteenth. He slides over a dirty martini, **and**	the date September eighteenth He slides over a dirty martini **glass**
and 'cheapest' therapist. Before long, he understood that, knowing nothing about the subject, it was hard to figure out which **therapist**	and cheapest therapist Before long he understood that knowing nothing about the subject it was hard to figure out which **one**
cups are too big to serve wine. "You didn't get half the things on my **cup**	cups are too big to serve wine You didn't get half the things on my **list**
is now going to introduce Watson to Sherlock in hopes that, um, Sherlock and, or, **you**	is now going to introduce Watson to Sherlock in hopes that um Sherlock and or **Watson**

Table 1: Examples of model input and predictions (blue if predicted token = true token, red otherwise).

text, but in many scenarios resource- or technical limitations make this unfeasible.

In this paper, we start addressing these questions by analyzing the language modeling behavior of OpenAI's GPT-2 (Radford et al., 2019) using a corpus of narratives available both as manually curated transcriptions and as noisier force-aligned transcripts. These manipulations make it possible to evaluate the impact of punctuation and casing removal on GPT-2's language modeling precision and uncertainty with very minimal preprocessing of the input text. By comparing next-token predictive accuracy and entropy across: a) parallel version of the corpus and b) multiple context sizes, we analyze how absence of these structural markers affects the model's ability to integrate information over longer text spans in order to formulate precise next-token predictions and reduce uncertainty.

2 Methods

2.1 Dataset

We evaluated GPT-2's language modeling behavior on next-token prediction using transcripts from the Narratives dataset (Nastase et al., 2021). The Narratives dataset, originally intended as a neural benchmark for models of language processing, includes transcripts from 27 thematically diverse audio narratives, and functional imaging (fMRI) data from participants listening to those narratives [2]. Transcripts are made available in three parallel versions: a manual transcript, cased and including punctuation (henceforth referred to as "baseline"); a cased, punctuation-stripped transcript; an uncased punctuation-stripped transcript produced by a force-aligned algorithm. Overall, each parallel version includes 42,989 words, and 1,440 of these are marked as "unknown" in the force-aligned transcript (the words not recognized by the force-alignment algorithm). These parallel versions of the corpus provide incremental manipulations of the presence of punctuation and casing (and an additional manipulation introducing lexical noise), while lexical content stays the same. To disentangle the effects of casing and lexical noise, we generated one more version of the transcripts, identical to the force-aligned transcription except for unknown tokens being replaced with lower-cased original tokens.

[2]Both can be accessed through DataLad (Halchenko et al., 2021) at `http://datasets.datalad.org/?dir=/labs/hasson/narratives`

2.2 Procedure

For each transcript type, we evaluated GPT-2 behavior in next-token prediction in a sliding window fashion, using a 1-word stride and different window sizes (5, 10, 15, 20, 25, 30, 50 words — where words are defined by whitespace boundaries). The manipulation in window size makes it possible to assess whether and how the model's ability to integrate information over longer contexts to produce more precise next-token predictions is affected by ablation of punctuation, casing, or addition of lexical noise. For each narrative and window size, the model iterates through corresponding chunks of text across all parallel corpora: at each iteration t, input to the model will include the same lexical context for all four corpora (with the exception of corrupted tokens). For a given window size s, words $w_t, w_{t+1}, ..., w_{t+s-1}$ are joined through whitespaces, tokenized, and fed to the corpus. w_{t+s} is tokenized, and the first of the resulting token is treated as true next token to compute performance metrics.

For each iteration t and each corpus, we extract a few predictive performance and uncertainty met-

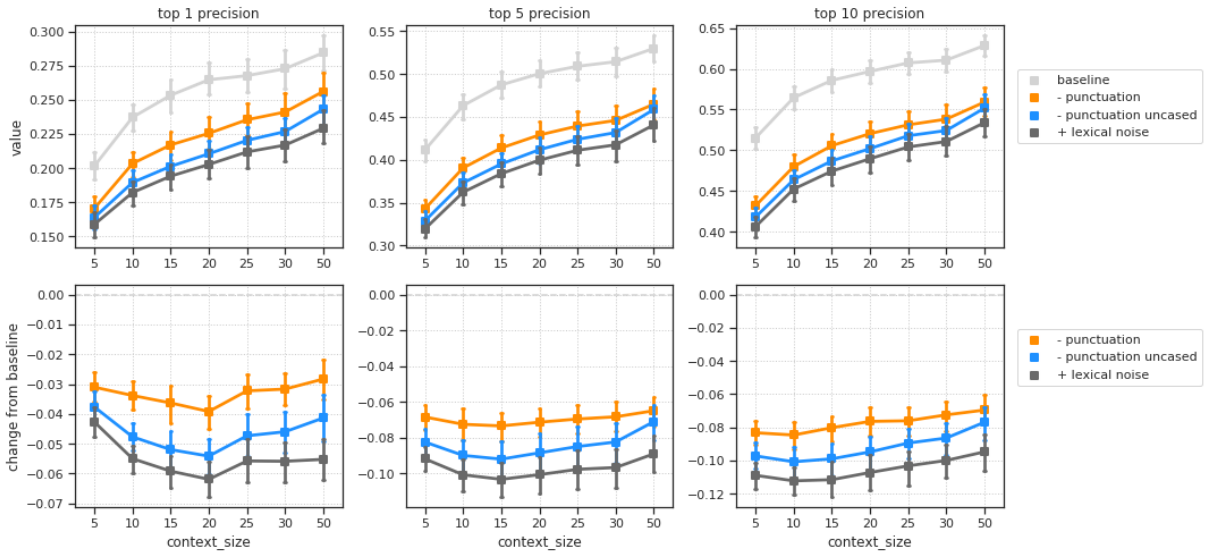

Figure 1: Proportion of cases (top: absolute values, bottom: difference from baseline) where the true word is assigned top probability (left), is among the tokens with the 5 highest probability scores (middle) or is among the tokens with the 10 highest probability scores (right), for each text type and context size. Error bars are 95% confidence intervals across narratives in the corpus.

rics. For performance, we focus on the model's precision in retrieving the true token (a more interpretable metric than cross-entropy loss). To quantify performance, we compute: a) a binary score quantifying whether the token with highest predicted probability is the true token (top 1 precision); b) a binary score quantifying whether predicted probability for the true token is one of the 5 highest predicted probability values (top 5 precision); c) a binary score quantifying whether predicted probability for the true token is among the 10 highest predicted probability values (top 10 precision). For uncertainty, we extract the entropy of the predicted probability distribution. To summarize the overall impact of punctuation, casing and lexical noise on the model's behavior, for each of these metrics we also compute correlations between values for the baseline transcript and values for each of the three manipulated versions.

3 Results

3.1 Precision

Overall, ablation of punctuation and casing and addition of lexical noise incrementally degrade precision.

Removal of punctuation contributes the most to a loss in precision (up to 4%, up to 6.5% and up to 8% for top 1, top 5 and top 10 precision respectively), while incremental casing and noise removal contribute to a smaller extent (up to 1% each for top 1 precision, and up to 2% each top 5 and top 10 precision). Overall, the model retains considerably good precision across manipulations (16-22% top 1, 35-45% top 5, and 42-53% top 10).

For all text types, precision systematically increase as context size increases, suggesting that absence of punctuation and casing does not hinder the models' ability to benefit from additional long-range information to refine its predictions. Qualitative inspection of model predictions suggests that, even when punctuation or casing are removed, the model generally produces plausible next-token predictions. Note that, for corresponding input sequences, predicted next tokens are often *different* across text types: the predicted token is the same across baseline and manipulated texts less than 10% of the time.

3.2 Uncertainty

All manipulations increase model uncertainty relative to the baseline, with punctuation having by far the largest effect. Interestingly, the effect of manipulations here interact with context size. When punctuation is available, the model benefits from the larger context to reduce its uncertainty. In absence of punctuation, however, entropy remains roughly constant across context sizes larger than 10 words (see Figure 2).

This effect is clarified by closer inspection of the predicted probability distribution (see Figure 4). In

the baseline, adding context increases probability mass in the head of the distribution, which reduces entropy. In absence of punctuation, as context size increases, probabilities remain roughly the same for the highest probability token (top left panel), and they decreases for its immediate competitors (top middle panel) and for highly implausible options (bottom right panel), but the countervailing increase in probability mass in the middle of the distribution (top right to bottom center panel) causes overall model uncertainty not to decrease.

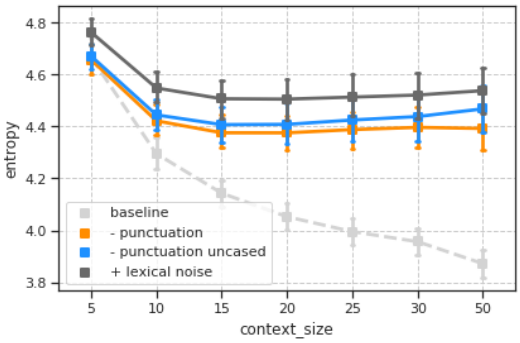

Figure 2: Entropy of the predicted probability distribution across text types and context sizes.

3.3 Overall similarity

Both next-token predictive performance metrics and entropy display medium to high correlations between baseline text and manipulated texts. Correlations range between .78 and .83 when punctuation is removed, between .72 and .77 when casing is removed, and between .69 and .73 when corrupted lexical tokens are added.

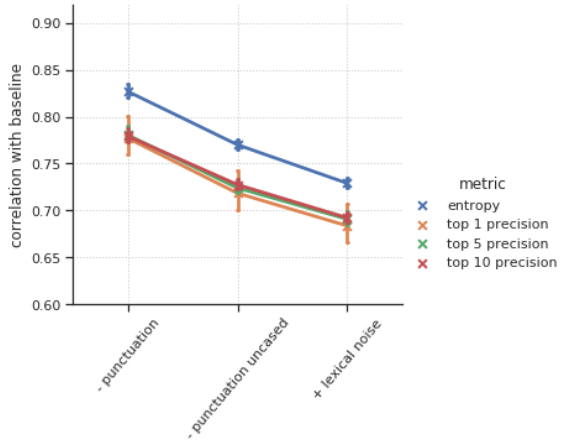

Figure 3: Correlations between baseline text and manipulated texts for both entropy and precision metrics.

4 Conclusions

We evaluated how manipulations of non-lexical markers (specifically, punctuation and casing) affects GPT-2's language modeling behavior. Absence of punctuation and casing increase uncertainty, and they decrease, but do not disrupt, model's ability to yield plausible language modeling predictions. Crucially, we observe that in absence of punctuation, GPT-2's precision increases when longer contexts are available, but — contrary to what observed for baseline text — longer contexts do *not* reduce uncertainty.

5 Limitations and future work

Our study provides a first contribution to understanding how transformers leverage structural and discourse information conveyed by non-lexical markers to perform language modeling predictions.

This study focuses uniquely on GPT-2, and the patterns observed in the present work may not generalize to other models. There are a number of factors that may modulate whether and how model behavior is significantly affected by the absence (or an inconsistent use) of non-lexical markers. Characteristics of the training corpus are one such example, with models trained on corpora including a larger proportion of unstructured text potentially being more robust than models trained mainly on highly structured text. Other relevant factors may include the mono- vs. multi-lingual nature of the model. Use of punctuation and casing is, in fact, far from consistent across languages. Multilingual models may therefore rely on non-lexical markers to a smaller extent compared to monolingual models. In a follow-up to this study, we are applying out evaluation pipeline to a wider range of pretrained models, including both models trained on forward language modeling and on masked language modeling, and including both monolingual and multilingual models.

The current study only evaluates the impact of non-lexical markers on *language modeling* performance. Yet, in most application scenarios, pretrained models are deployed in the context of downstream tasks (e.g., classification). Future iterations of this work will combine an evaluation of the effect of removing non-lexical markers on language modeling behavior with an evaluation of its impact on common downstream tasks.

Finally, this study compares GPT-2's behavior across scenarios where non-lexical markers are ei-

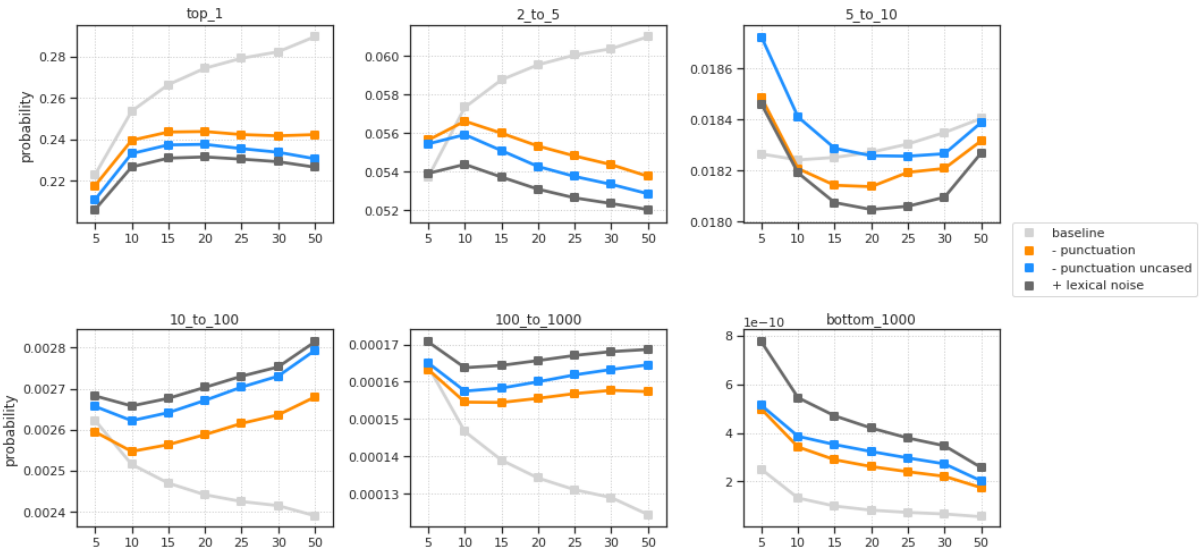

Figure 4: Average probability for the top value in the distribution (top left), 2nd to 5th top values (top middle), 5th to 10th top values (top right), 10th to 100th top values (bottom left), 100th to 1000th top values (bottom centre) and bottom 1000 values (bottom right).

ther present and used consistently or fully absent, but there are several (and perhaps more realistic) scenarios in between. Future work will also target these intermediate scenarios, using a more varied set of corpora or probabilistic text augmentation.

References

John Austin. 2019. The Book of Endless History: Authorial Use of GPT2 for Interactive Storytelling. In *Interactive Storytelling*, Lecture Notes in Computer Science, pages 429–432, Cham. Springer International Publishing.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *arXiv:2005.14165 [cs]*. ArXiv: 2005.14165.

Maury Courtland, Adam Faulkner, and Gayle McElvain. 2020. Efficient Automatic Punctuation Restoration Using Bidirectional Transformers with Robust Inference. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 272–279, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*. ArXiv: 1810.04805.

Yaroslav O. Halchenko, Kyle Meyer, Benjamin Poldrack, Debanjum Singh Solanky, Adina S. Wagner, Jason Gors, Dave MacFarlane, Dorian Pustina, Vanessa Sochat, Satrajit S. Ghosh, Christian Mönch, Christopher J. Markiewicz, Laura Waite, Ilya Shlyakhter, Alejandro de la Vega, Soichi Hayashi, Christian Olaf Häusler, Jean-Baptiste Poline, Tobias Kadelka, Kusti Skytén, Dorota Jarecka, David Kennedy, Ted Strauss, Matt Cieslak, Peter Vavra, Horea-Ioan Ioanas, Robin Schneider, Mika Pflüger, James V. Haxby, Simon B. Eickhoff, and Michael Hanke. 2021. DataLad: distributed system for joint management of code, data, and their relationship. *Journal of Open Source Software*, 6(63):3262.

Danny Merkx and Stefan L. Frank. 2020. Human Sentence Processing: Recurrence or Attention? Technical report. Publication Title: arXiv e-prints ADS Bibcode: 2020arXiv200509471M Type: article.

Samuel A. Nastase, Yun-Fei Liu, Hanna Hillman, Asieh Zadbood, Liat Hasenfratz, Neggin Keshavarzian, Janice Chen, Christopher J. Honey, Yaara Yeshurun, Mor Regev, Mai Nguyen, Claire H. C. Chang, Christopher Baldassano, Olga Lositsky, Erez Simony, Michael A. Chow, Yuan Chang Leong, Paula P. Brooks, Emily Micciche, Gina Choe, Ariel Goldstein, Tamara Vanderwal, Yaroslav O. Halchenko, Kenneth A. Norman, and Uri Hasson. 2021. The "Narratives" fMRI dataset for evaluating models of naturalistic language comprehension. *Scientific Data*, 8(1).

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners.

Martin Schrimpf, Idan Asher Blank, Greta Tuckute, Carina Kauf, Eghbal A. Hosseini, Nancy Kanwisher, Joshua B. Tenenbaum, and Evelina Fedorenko. 2021. The neural architecture of language: Integrative modeling converges on predictive processing. *Proceedings of the National Academy of Sciences*, 118(45). Publisher: National Academy of Sciences Section: Biological Sciences.

Karsten Steinhauer. 2003. Electrophysiological correlates of prosody and punctuation. *Brain and Language*, 86(1):142–164.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Andris Vāravs and Askars Salimbajevs. 2018. Restoring Punctuation and Capitalization Using Transformer Models. In *Statistical Language and Speech Processing*, Lecture Notes in Computer Science, pages 91–102, Cham. Springer International Publishing.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books. *arXiv:1506.06724 [cs]*. ArXiv: 1506.06724.

Piotr Żelasko, Raghavendra Pappagari, and Najim Dehak. 2021. What Helps Transformers Recognize Conversational Structure? Importance of Context, Punctuation, and Labels in Dialog Act Recognition. *Transactions of the Association for Computational Linguistics*, 9:1163–1179.

A Appendix

original transcript. Jerry and George strolled through the airport with their suitcases. George walked quickly, grimacing as he scanned the signs to figure out which way to go. A man passing by sneezed in his direction, causing him to recoil backwards and then frantically squirt Purell onto his hands.
- punctuation. Jerry and George strolled through the airport with their suitcases George walked quickly grimacing as he scanned the signs to figure out which way to go A man passing by sneezed in his direction causing him to recoil backwards and then frantically squirt Purell onto his hands
- casing jerry and george strolled through the airport with their suitcases george walked quickly grimacing as he scanned the signs to figure out which way to go a man passing by sneezed in his direction causing him to recoil backwards and then frantically squirt purell onto his hands
- casing noised jerry and george strolled through the airport with their suitcases george walked quickly <unk> as he scanned the signs to figure out which way to go a man passing by sneezed in his direction causing him to <unk> backwards and then frantically squirt <unk> onto his hands jerry <unk> up

Table 2: Sample excerpts from different transcript types

text type	input	next word	true token	predicted
manual transcript	their suitcases. George walked quickly, grimacing as he scanned the signs to figure out which way to go. A man	passing	pass	in
- punctuation	their suitcases George walked quickly grimacing as he scanned the signs to figure out which way to go A man	passing	pass	in
- casing	their suitcases george walked quickly grimacing as he scanned the signs to figure out which way to go a man	passing	pass	in
- casing noised	their suitcases george walked quickly <unk> as he scanned the signs to figure out which way to go a man	passing	pass	was

Table 3: inputs to the model, next word, true token, and model predictions for window size 20.