Sentiment Bias in Predictive Text Recommendations Results in Biased Writing

Kenneth C. Arnold[1] Krysta Chauncey[2] Krzysztof Z. Gajos[1]
[1]Harvard School of Engineering and Applied Sciences
Cambridge, MA, USA
{kcarnold, kgajos}@seas.harvard.edu
[2]Draper
Cambridge, MA, USA
kchauncey@draper.com

Abstract

Prior research has demonstrated that intelligent systems make biased decisions because they are trained on biased data. As people increasingly leverage intelligent systems to enhance their productivity and creativity, could system biases affect what people create? We demonstrate that in at least one domain (writing restaurant reviews), biased system behavior leads to biased human behavior: People presented with phrasal text entry shortcuts that were skewed positive wrote more positive reviews than they did when presented with negative-skewed shortcuts. This result contributes to the pertinent debate about the role of intelligent systems in our society.

Index Terms: Human-centered computing—Human-computer interaction

1 Introduction

Interactive intelligent systems promise to help us complete our tasks faster and more efficiently. By predicting our desires, they can adapt their presentation or behavior in ways that make it easier to bring about what we desire. For example, complex applications can predict relevant toolbar items and present them in a quick access toolbar [13], photo manipulation tools can predict the desired image from a small number of manipulations by the user [36], and predictive typing systems guess the words or phrases we want and show them as "suggestion" buttons for quick insertion [5, 20, 28].

What effect do these systems have on what people create using them? Most of the systems mentioned have been evaluated on efficiency criteria, such as speed and accuracy with which a person can transcribe given text. But the process of interacting with these intelligent systems may shape our actions, creative artifacts, and desires in unexpected ways. For example, adaptive interfaces can hinder users' awareness of unused features of the application [13]. Music recommender systems make music tastes converge [15]. And use of predictive typing systems may affect spelling and morphological skills [34].

Since data-driven prediction is a central element to how many intelligent systems help users, we are inspired by recent studies of showing conditions under which machine learning systems exhibit prejudiced behavior. Although machine learning algorithms do not contain any discriminatory biases by themselves, recent work has demonstrated that systems based on these algorithms can make prejudiced decisions—in domains such as hiring, lending, or law enforcement—if the data sets used to train the algorithms are biased [2]. Such biased data sets are more common than initially suspected: Recent work demonstrated that two popular text corpora, the Google News dataset and the Common Crawl database of website text, contain race and gender biases, and machine learning systems incorporate those biases into their internal representations [8] unless specific effort is made to remove a given bias [7]. For this paper, we will use the term "bias" to refer to any systematic favoring of certain artifacts or behavior over others that are equally valid.

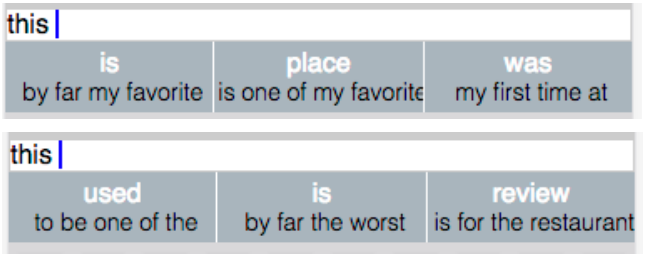

Figure 1: Biases in training data can cause intelligent systems to offer assistance that is unintentionally biased. In our experiment, we manipulate a predictive text system to exhibit a bias towards positive sentiment (top screenshot) or negative sentiment (bottom screenshot). Although the stated purpose of showing these predictions is efficiency, do biases in the prediction content affect the sentiment of what people write?

Motivated by the interest in understanding and mitigating bias in intelligent systems, we asked two questions: (1) can intelligent systems that support creative tasks exhibit unintentional biases in the content of the support that they offer, and (2) do these biases affect what people produce using these systems?

We investigated these questions in the context of a commonly used intelligent interactive system, namely, predictive typing on mobile devices. We focused on the task of writing restaurant reviews, an everyday task that people often do on mobile devices. Writing can exhibit many kinds of biases, such as race, gender, or culture; we focused on one type of bias: the valence of the sentiment that is expressed in a review, i.e., is the review favorable or unfavorable toward the restaurant? In addressing the first question, we found that available review corpora are biased towards positive content, and that a standard text-generation algorithm generated recommendations that humans perceived as biased positive, even after rebalancing the dataset using star ratings. Then, by manipulating the sentiment of recommendations in a controlled experiment, we found that positively biased recommendations bias people to write content that is more positive. Taken together, these two studies suggest a *chain of bias*: biases in training data cause biases in system behavior, which in turn cause biased human-generated products.

Our contributions are:

- Evidence that naive text prediction systems for review-writing domains can produce recommendations that are biased towards positive sentiment
- A method for shaping the sentiment of contextual recommendations generated during real-time typing.
- A study demonstrating that writers generate restaurant reviews

To appear in Proceedings of Graphics Interface 2018

with more positive content when presented with positive recommendations.

2 Background and Related Work

Effective use of predictive text requires that writers frequently read the set of recommended words, the content of those recommendations may influence writers' goals. Writers' goals are generally not tightly prescribed but instead develop during the process of composition [31]. For example, writers plan sentences only a few words ahead of producing them [25], pause longer between semantic units of different size [31], and look back on previously written text while writing new text [31, 32]. A recent study finds that the name "suggestion bar," used in prior literature to refer to these recommendations [28], may be apt: writers use the predicted words not just when they match a specific word already in mind, but conform their writing goals to incorporate those words [1].

The sentiment of the recommendations could affect the sentiment of the result through several different mechanisms. First, the recommended phrases may provide specific positive (or negative) information that the writer uses, either because it is easy to enter via accepting the recommendation verbatim, or because the recommendation reminds them of an aspect to discuss even if they do not use the exact words recommended. Second, the recommendations may "prime" the writer as they implicitly function as examples of what sentiment of writing is expected: if all examples are positive, a writer may feel like a negative phrase is out of place; in contrast, recommendations with a diversity of sentiments may convey that a variety of sentiments is expected.

Evaluations of systems that are explicitly designed as creativity support tools often consider how interaction with the system affects the result of what is created. For example, participants using the *Adaptive Ideas* system were found to produce web designs that were rated more highly by external raters than those produced by participants using a baseline system [22]. The *Painting with Bob* system was designed and evaluated for its impacts on novices' creative processes [3]. The authors highlight personal style as a "vital aspect of creative expression." However, they hold this goal in tension with "creative flexibility," since the assistive characteristics of the system can also be constraints.

Text recommendation is prevalent in interactive systems today, both in research and in deployed systems. Besides predictive typing on mobile devices, previously developed systems also predict text to assist search query formulation [11], suggest complete responses to messages [19], recommend improvements to grammar or style [26], and show examples to assist learners and non-native speakers [9, 10].

Recent research has brought large improvements in systems' abilities to model existing natural language. The perplexity of a language model measures its uncertainty about what will be said; perplexities of state-of-the-art language models have improved from 67.6 to 23.7 on one popular benchmark [17]. Further benefits can be achieved by leveraging context from images or other modalities [18, 23] and dialogue [33].

Less work has been done in being able to adjust interpretable aspects of the generated text. Review metadata can be used to generate text with various sentiment [24], and manipulating the internal state of some models has been shown to also adjust sentiment [29]. Explicit representation of desired attributes together with adversarial training can allow the generation to be controlled along other desired aspects also [16].

3 Study 1: Recommendations are Biased Positive

Predictive typing recommendations can vary in the sentiment that they express. For example, consider a writer composing a restaurant review. After the writer types 'The', the system could choose to recommend 'best' (positive sentiment valence), 'worst' (negative valence), or 'food' (neutral). The strength of sentiment expressed in recommendations can be even stronger when the system can recommend phrases (e.g., "staff are super friendly," "prices are very reasonable," or "only good thing about").

Dataset	N	Median	Mean $\pm$ stdev
Yelp	196,858	4	3.59 ± 1.17
Amazon	82,456,877	5	4.16 ± 1.26
TripAdvisor	1,621,956	4	3.93 ± 1.21

Table 1: Examples of readily available datasets of reviews with star ratings: number of reviews (N) and statistics of their star ratings (1–5, 5 highest). Review datasets are biased positive. Datasets: Yelp (restaurants only, from `https://www.yelp.com/dataset`) Amazon product reviews (from `http://snap.stanford.edu/data/amazon/productGraph/`), TripAdvisor (from `http://sifaka.cs.uiuc.edu/~wang296/Data/index.html`)

Are contemporary predictive typing systems equally likely to offer positive recommendations as negative recommendations, or do they exhibit biases in sentiment? We first study the biases present in existing approaches for recommendation in writing.

3.1 Online review corpora have biased sentiment

Suppose a practitioner wants to develop an intelligent system for supporting review-writing. This system likely needs a set of existing reviews so that it can learn typical characteristics of reviews. A reasonable strategy would be to search the Internet for datasets of user-generated reviews and pick one with an appropriate domain and size. Unfortunately, this reasonable strategy is likely to give the practitioner a biased dataset. Table 1 shows the distribution of star ratings in several corpora of online reviews. In none of these readily available review datasets is the mean or median star rating below a 3 on a 1–5 scale. Though these are far from the only collections of review texts on which a practitioner may train a text generation system, their bias is clear and large, even when only considering star rating as a coarse proxy for overall sentiment.

3.2 Systems trained on review corpora make biased recommendations

Predictive typing systems use statistical models of language to estimate the probability that a given word or phrase will follow in a particular context, then show the phrases with the highest probability. Consider a corpus composed of several different groups, for example, positive, neutral, and negative restaurant reviews. If the training data contain more examples of one group than the others, then the predictions will favor a relevant word or phrase from the more common group over an equally relevant word or phrase from a less common group simply because that phrase occurs more frequently.[1]

In this section, we demonstrate that phrase prediction systems do present biased recommendations to writers.

Recommendation Generation System We used a phrase recommendation generation system similar to [1]. The system uses a 5-gram Kneser-Ney language model [14] trained on restaurant reviews from the Yelp Academic Corpus. The system generates contextual recommendations in two steps: first, it selects the three most likely next-word predictions, then it generates the most likely phrase continuation for each word using beam search.

To try to correct for the overabundance of positive reviews that we noted above, we randomly subsampled the Yelp corpus so that there were an equal number of reviews with each of the five available star ratings. We also held out a 10% sample of reviews for validation

[1] A second cause of stereotyped predictions is more subtle: even if the probabilities could be corrected for the differences in base rates between groups, the *accuracy* of the model will be lower in underrepresented groups because of the reduced amount of training data.

experiments described below. Despite the smaller training set size, the relevance of the recommendations seemed qualitatively sufficient for our purposes. We will refer to this text generation system as BALANCED in this paper, though as the results below show, the system's output is not actually balanced.

Re-typing Paradigm We simulated re-typing existing reviews and generated recommendations using BALANCED, then compared the recommendations with the text that was originally typed. We constructed samples to evaluate in the following way. First we subsampled held-out reviews evenly from the 5 star rating classes. For each review, we picked a word boundary such that at least 5 words remained before the end of the sentence. We then simulated retyping the review until that word boundary and extracted the set of recommendations that the system would present. We picked one of the 3 recommendations uniformly at random and presented it in comparison with the 5 words that actually followed in the original review. If the recommendation happened to match the original text exactly, we drew a new location.

Writing process theories posit that writers pause longer at sentence boundaries than other word boundaries because they are planning their next sentence [31]. While doing so, they often read what they have already written [32]. Thus, a recommendation displayed at the beginning of a sentence has a larger potential impact on the writer's plan for the sentence that follows. Since the retyping process described above would otherwise sample sentence beginnings rarely, we oversampled sentence beginnings by deciding uniformly whether to pick the beginning of a sentence or a different word.

Recommendations trend positive, especially phrases We compared the sentiment of the text of the recommendations offered by the system with the text that was actually written in the original review. To do so, we presented pairs of texts (with their original context of 5 prior words) to MTurk workers and asked which they perceived as more positive. Workers could also choose a "neither" option if the sentiment valence was indistinguishable. The interface randomized whether the recommendation or the original text was shown first. We showed each pair to 3 different workers and took the majority vote (breaking three-way ties in favor of "neither"). We coded the result as an ordinal variable taking values -1 (original word/phrase selected as more positive), 0 (neither), or 1 (recommended phrase selected as more positive).

We collected rankings for 500 recommendations. A binomial test showed that at sentence beginnings, the recommendations were picked as more positive significantly more often (164 out of 263 total recommendations, $p < .0001$) than the original review text, across all star ratings of original reviews. For mid-sentence recommendations, the difference was less pronounced, but in comparisons where there was a winner (rather than "neither"), the generated text was more positive than the original text significantly more often (112 out of 184 decided comparisons, $p = .003$).

Since the original star rating of the review should predict how positive the original text is, we expected it to influence how its sentiment compares with the generated text. If the generated text were always consistently like that of a 5-star review, we would expect a strong influence of star rating on the binary comparison: the original text would always be more positive than text from 1-star reviews, but compared with text from 5-star reviews it would be a toss-up. On the other hand, if the generated text tended to follow the sentiment of the original text (because the context of the recommendation leads in a particular direction), the star rating would have a relatively minor effect on the binary comparison.

Figure 2 shows that generated phrases were rated on average more positive than the original text, but less so for higher star ratings and for mid-sentence recommendations. To quantify this effect, we fit two separate ordinal logistic models predicting the more positive option, one for beginning-of-sentence recommendations and one for mid-sentence recommendations, with the star rating of the original review as an ordinal fixed effect. For beginning-of-sentence recommendations, we observed a strong effect of review star rating: the likelihood ratio was 31.7, $p < .0001$. For mid-sentence recommendations, we observed a much weaker effect (likelihood ratio of 9.79, p=0.044).[2]

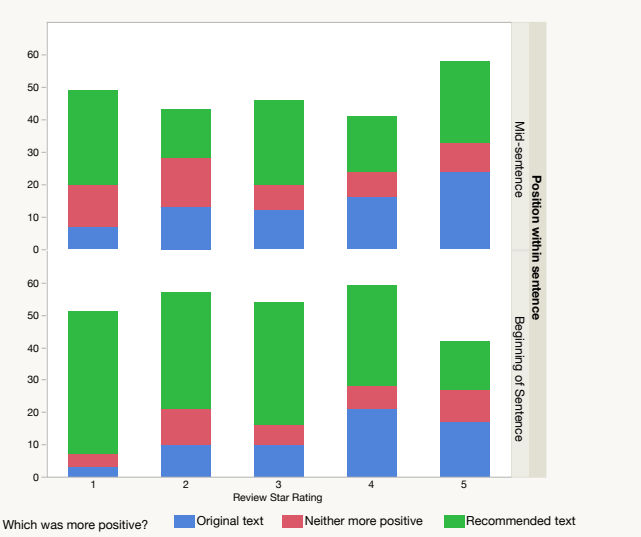

Figure 2: Stacked histograms of the number of times that phrases generated by the BALANCED system were chosen by crowd workers as more positive than naturally occurring text, in pairwise comparisons. The generated text was usually perceived as more positive, reflecting a bias towards positive content in the training data. The effect was strongest for recommendations at the beginning of a sentence (lower panel).

These findings indicate that generated phrases at the beginning of sentences were more strongly positive than what people wrote without those recommendations. In the middle of a sentence, recommendation sentiment stayed closer to the sentiment in the original text, but still leaned positive.

These findings were in the context of *phrase* recommendations. To determine if single-word recommendations are also perceived as biased, we repeated this entire process of recommendation sampling (with a different random seed) and annotation to generate another 500 recommendation pairs, except that this time we limited both the original and generated text to a single word. We found that single-word recommendations did not have a clear difference in sentiment compared with the corresponding original words: in most beginning-of-sentence pairs, participants indicated that neither text was more positive; mid-sentence votes were split evenly among the three options.

3.3 Discussion

The above findings indicate that phrase recommendations reflect the positive bias observed in the training data: generated phrases are usually perceived as more positive than the text actually written in prior reviews.

It is not clear why a system trained on a dataset with equal counts in each star rating would be biased positive. If positive reviews tended to be longer than negative reviews, that could explain the bias, but in fact negative reviews tend to be longer (146.6 words for 1-star

[2]Since the effect size for ordinal effects in ordinal regressions is unintuitive, we repeated the analysis with the original star rating as a continuous effect. For beginning-of-sentence recommendations, the log-odds was 0.43 per star (95% CI 0.304–0.691), and for mid-sentence recommendations, the log-odds was 0.18 (CI 0.019–0.349).

reviews vs 115.4 words for 5-star reviews). A possible explanation is that even 1-star reviews often have some characteristics of positive content, such as phrases like "a friend recommended this to me." Also, some negative reviews start off with positive aspects of the product or experience before beginning their complaints.

4 Study 2: Effect of Recommendation Sentiment on Writing Artifacts

We have found that biased training data results in biased recommendations; we now study whether biased recommendations lead to biased writing output. In this section we describe an experiment in which we present writers with recommendations manipulated to vary in sentiment valence, and measure the effects that these recommendations have on the sentiment of the resulting writing. In this section, we first introduce the interactive system that we build on, then discuss the conceptual design of the experiment, and the modifications needed to manipulate the behavior of the system. We then describe the details of the experimental task, measures, and procedure.

4.1 Interactive System for Text Recommendation

The intelligent interactive interface that we use for this experiment is a phrase-shortcut keyboard [1]. The UI shows the word in the familiar "suggestion bar" interface used in contemporary mobile phone keyboards, to insert or complete a word in a single tap. The system also offers a phrase continuation, which is shown as a preview below the word, indicating that subsequent taps on the same suggestion box will insert subsequent words from that phrase. This feature can increase efficiency because if writers notice that part or all of the phrase communicates what they want to say, they can insert many words quickly by using a repeated-tap gesture. The system updates the phrase recommendations after each keypress or tap, under the constraint that if a recommendation was tapped, the new recommended phrase must start with the phrase that was previously shown in that suggestion box. If a word has only partially been entered, the recommendations offer completions of that word, otherwise the recommendations predict a likely next word.

4.2 Experiment Design

We hypothesize that when writers are given positive recommendations, their writing will include more positive content than when they are given negative recommendations. To test this hypothesis, we needed to manipulate the sentiment of recommendations that a system provides to participants and measure the sentiment valence of their writing. It is not possible to offer recommendations that are uniformly "positive" or "negative"; in the middle of a glowingly positive sentence, a negative recommendation would be seen as irrelevant; in a purely factual sentence, it may not be possible to offer text that has any perceived sentiment at all. Instead, we *skew* the distribution of sentiment of the generated text: in the condition we call SKEW-POS, we increase the likelihood that the system generates a positive recommendation instead of a negative one, and in SKEW-NEG, we increase the corresponding likelihood of negative recommendations. As the results of Study 1 suggest, the differences between these two systems will be most apparent at the beginning of a sentence. The system must manipulate the sentiment of the recommendations without being irrelevant, ungrammatical, or unreasonably extreme.

Since we expected that participants may take some time to react to changes in recommendations, we chose to keep the recommendation strategy constant for each writing artifact (in this case, a restaurant review), changing only between artifacts. Since we expected individual differences in behavior and artifact, we used a within-subjects design and mixed-effects analysis. We had participants write about both positive and negative experiences, for a total of 2 (prior sentiment valence) x 2 (recommendation valence) = 4 trials for each participant. In our analyses, we fit random intercepts for each participant and include block and prior sentiment as ordinal control variables, unless otherwise noted.

4.3 Manipulating Sentiment of Recommendations

Controlling the sentiment of text generation is an active area of research [16, 24, 29]. However, we were not yet able to get these new techniques to run at interactive speed on commodity hardware. On the other hand, training on only reviews of a certain star rating unduly compromised the relevance of the language model. So for the present experiment, we used a simple "reranking" approach in which a contemporary language generation system generates a large number of candidate phrases, then a second process re-orders these candidates to pick the most positive (for SKEW-POS) or negative (for SKEW-NEG).

The system generates the set of candidate phrases using a modification of the beam-search process used in the system of study 1. We used the same base language model as for that study, BALANCED, based on subsampling the Yelp review corpus so that it had an equal number of reviews with each star rating (1 through 5). However, we modified the beam search process so that it would generate a range of possible phrases. To generate candidate phrases, the system first identified the 20 most likely next words under the language model, then for each word generated the 20 most likely phrases starting with that word using beam search with width 20, resulting in 400 candidate phrases.

We then used a classifier to select a set of phrases from among the candidate set according to the desired sentiment skew. We trained a Naive Bayes classifier to predict review star rating using bigrams as features.[3] For each candidate phrase, the system computed the probability that each phrase came from a 5-star review (for SKEW-POS), or a 1-star review (for SKEW-NEG). A simplistic approach would be to then recommend the phrases with the most extreme positive (or negative) sentiment. However, the phrases with the most extreme sentiment were sometimes ungrammatical, awkward, or simply irrelevant. We found that pilot study participants tended to ignore recommendations that they perceived as irrelevant, so we added a likelihood constraint to the generation process: the system first picked the three phrases with highest likelihood under BALANCED that start with distinct first words, then iteratively replaced each phrase with one of the candidate phrases that was more positive (or more negative), so long as (1) the set of recommendations would still start with distinct first words and (2) the contextual likelihood of the replacement phrase was no less than β times the likelihood of the phrase it replaced. We chose $\beta = e^{-1} \approx 0.36$ (one nat) because the resulting phrases tended to be grammatically acceptable and still skewed in sentiment. Although likelihood can be a poor proxy for grammatical acceptability [21], the approach seemed reasonably successful in pilot studies. Figure 1 shows an example of the output of this approach. We parallelized language model evaluations in the beam search to increase speed. The overall latency from tapping a key to seeing a recommendation was typically less than 100ms with commodity hardware, which is similar to the latency of deployed smartphone keyboards.

4.4 Validation of the sentiment manipulation approach

We validated our sentiment manipulation approach using the sentiment analysis functionality of Google Cloud NLP.[4] Using a methodology identical to that used in Study 1, we generated 500 sample contexts. We took the last 6 words of each context and appended each of four phrases: the text that had followed that context in

[3] For short snippets, such as the phrases that we evaluate, such a simple approach can outperform more complex models [35].

[4] https://cloud.google.com/natural-language/docs/analyzing-sentiment

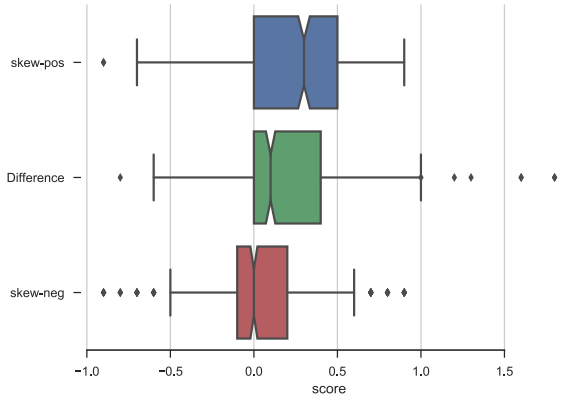

Figure 3: Box plots of sentiment scores computed by the Google Cloud NLP sentiment API for recommendations generated by SKEW-POS (top; mean score 0.23) and SKEW-NEG (bottom; mean score 0.07), for the same 500 phrase contexts drawn from existing reviews; middle plot shows the difference (SKEW-POS - SKEW-NEG) for each context. Notches show confidence intervals for the median. The sentiment manipulation method successfully creates a difference in sentiment valence. Neither system was always positive or always negative, however.

the original review (TRUE), and the recommendations[5] generated by each of the three systems we studied: the baseline BALANCED system (used in Study 1), the SKEW-POS system, and the SKEW-NEG system. We then submitted each of the 500*4=2000 resulting phrases to the Google Cloud NLP sentiment analysis service and recorded the sentiment score (which ranges from -1.0 for maximally negative to +1.0 for maximally positive).

Figure 3 shows that the sentiment manipulation approach successfully created a difference in sentiment valence between the SKEW-POS and SKEW-NEG systems. An all-pairs Tukey's HSD test confirms that the difference in sentiment means between SKEW-POS, SKEW-NEG, BALANCED, and TRUE was significant at the p=0.05 level for all pairs except for SKEW-POS and BALANCED. Note, though, that all means are above 0.0, indicating that in no condition are the recommendations more negative than positive on average.

4.5 Task

We asked participants to write restaurant reviews using touchscreen keyboards that offered word and phrase shortcut entry functions. We modeled our task design on the design used in [1]. The reviews that we asked participants to write were about specific experiences at actual restaurants that they committed to write about before the experiment began. Our instructions motivated accuracy and quality using quotes from the reviewing guidelines on Yelp[6] and promised a bonus for high quality reviews. We also encouraged participants to avoid typos, since the contextual recommendation system relied on accurate spelling of the context words.

4.6 Procedure

We implemented a simplified touchscreen keyboard as a mobile web application using React, using WebSockets to connect to a server that generated recommendations and managed experiment state. After tapping on a recommended word, the web application opportunistically advanced the recommendation to the next word in the corresponding phrase, so multiple words from a phrase could be entered quickly, without waiting for a response from the server. The keyboard was designed to mimic a standard mobile phone keyboard in look and feel, but simplified to be modeless. As such, it only supported lowercase letters and a selected set of punctuation. For simplicity and to focus attention on the prediction (rather than correction) aspect of the typing interface, the keyboard did not support autocorrect.

We recruited 38 participants from a university participant pool to participate in our web-based study. Participants were compensated with a gift card for $12 for an estimated duration of 45–70 minutes. Study procedures were approved by the university IRB. Participants were instructed to use their own mobile devices, so screen size and device performance varied between participants.

At the start of the experiment, we asked participants to list 4 establishments that they would like to write about, two above-average experiences and two below-average experiences. For each one, we also asked for their overall opinion about the establishment in terms of a star rating. We chose this procedure so that participants would be strongly encouraged to report faithfully about their experiences with accuracy and detail, rather than making up an imaginary review in order to play with the recommendations or get through the experiment quickly.

Participants then completed a tutorial to familiarize themselves with the keyboard and recommendations. Participants were instructed to write a sentence about the interior of a residence they know well, as if writing a description for a site like Airbnb. During the tutorial, the keyboard presented recommendations using the same algorithms as the main experiment, but with training data drawn from Airbnb postings from 16 cities in the US[7] and without sentiment manipulation.

The system then instructed participants to write about the four establishments they listed, one at a time. To ensure that each participant experienced all four combinations of writer sentiment and recommendation sentiment, the order of establishments was randomized in a specific way. First the system chose whether to have the participant write about the above-average experiences or below-average experiences first, then it shuffled the restaurants within each category. The order of conditions was also randomized: the first condition is randomly chosen as one of SKEW-POS or SKEW-NEG, then subsequent conditions alternated.

The framing used to describe the recommendations is important to the validity of our experiment. A term such as "recommendation" or "suggestion" carries an implication that the content of the recommendations reflect what the experimenter desires. If participants simply viewed the recommendations as telling them what they should write, then the effect of recommendations on writing content would be trivial. Even with more neutral language such as "words or phrases offered by the keyboard," participants may still make a guess at the intent of the researchers. Instead, we needed to actively focus participants on a different aspect of the recommendations. Since the selling point of these systems is usually efficiency, we chose to emphasize that aspect. We did this in two ways: first, we referred to the recommendations as "shortcuts." Second, we added a feedback mechanism to help participants gauge whether the recommendations would help them write more efficiently. Since the recommendations offered by our system were generally much more relevant to the task than the domain-general recommendations that participants may have been accustomed to from their experience with predictive typing keyboards, we added a feedback element to the interface: whenever a participant typed a character that caused the current word to be a prefix of one of the words that is currently being presented as a recommendation, the interface highlighted that fact: that word remained in its corresponding recommendation slot (even if the recommendations generated after entering new character would have

[5] We randomly selected one of the three recommendations that would have been shown.

[6] https://www.yelp.com/guidelines

[7] Data from `http://insideairbnb.com/get-the-data.html`.

otherwise caused it to be reordered), the recommendation flashed, and the prefix was highlighted.

After each of the four trials, participants completed a short survey asking what star rating they would now give to the establishment, and how the sentiment of the "shortcuts" compared with the experience they were writing about.

4.7 Measuring Sentiment of Writing Artifacts

For evaluating the sentiment of complete writing artifacts, we chose the unit of analysis to be the sentence: smaller units would be tedious and potentially ambiguous (e.g., for "the only good thing about this place," what is the sentiment of "good"?); larger units such as paragraphs or complete writings are overly coarse. Since each sentence can contain both positive and negative content (e.g., "the service was atrocious but the food made up for it") or neither (e.g., "each entree comes with two sides") [4], we asked annotators to rate, for each sentence, how much positive content it had and how much negative content it had. Pilot rating studies showed that raters could only reliably distinguish three levels of sentiment content, so we had annotators rate the positive content of each sentence on a scale of 0 (no positive content), 1 (vaguely positive content), or 2 (clearly positive content), and the negative content of each sentence on a corresponding scale. ([27] reports similar limitations of annotation granularity.) We computed the mean across raters for each sentence. We used the `sent_tokenize` routine from NLTK [6] to split reviews into sentences. We summarized the sentiment of a review by two quantities: the mean amount of positive sentiment and mean amount of negative sentiment, taken across sentences.

4.8 Adjustments to Data

Despite instructions to avoid typos, most reviews included one or two clear typos. (Recall that the keyboard did not employ autocorrect.) Since interpretation of typos can be difficult and sometimes ambiguous for annotators, we added a typo correction step before sentiment annotation. The typo correction was done by one of the authors, blind to all metadata, with the assistance of the Microsoft Word contextual spelling checker. In almost all cases the intended text was unambiguous; the few ambiguous cases were left as-is. We did not exclude participants based on excessive typos.

Despite our instructions to list two positive and two negative experiences to write about, and separate labeled areas to enter positive and negative experiences, some participants did not list any experience with less than 4 stars out of 5. So while we used the participant's labeling of experiences as positive and negative to determine the trial and condition order (as described in the Procedure section above), and had planned to use that label as a control variable in our analysis, because of this mismatch we decided to use their star rating for analysis instead. Nevertheless we have reason to believe that the counterbalancing was successful: a regression of star rating on condition has an r^2 of less than 0.01.

5 Results

5.1 Effects on Writing Artifacts

We recruited annotators from MTurk to perform the task described in Sect. 4.7 to measure the sentiment of complete writing artifacts. Each annotator rated one or more batches of 4 writing samples, randomly chosen from among the entire set of writing samples. Each of the 38*4=152 writing samples was rated by three annotators. Krippendorf's alpha agreement was 0.84 on positive sentiment and 0.85 on negative sentiment, indicating acceptable agreement.

We observed a significantly larger amount of positive sentiment in the reviews written with positive-skewed recommendations (SKEW-POS condition M=1.22, σ=0.67) compared with negative-skewed recommendations (SKEW-NEG condition M=0.98, σ=0.61) ($F_{1,106.8} = 12.3$, p=.0007). For comparison, the magnitude of the effect of switching from SKEW-NEG to SKEW-POS is 77% of the estimated magnitude of having given one additional star.[8]

We did not observe a significant difference between conditions in the amount of negative sentiment in the reviews written ($F_{1,107.4} = 0.85$, n.s.). Since the validation showed that the SKEW-NEG condition was only relatively negative, not negative in an absolute sense, this result is not surprising.

Compared to the star rating that participants gave their experiences when listing them at the start of the experiment, participants gave an average of 0.27 more stars to their experience after writing about it in the SKEW-POS condition, and 0.1 more stars after the SKEW-NEG condition. However, a mixed ANOVA did not show a statistically significant effect of condition ($F_{1,98.13} = 1.55$, n.s.).[9]

These results reflect analyses that included all participants. We observed that a few participants typed their reviews almost exclusively by tapping recommendations. Though this may have been honest behavior, it seems more likely that it was done in an attempt to complete the experiment with minimal effort, or a misinterpretation of the instructions. We re-ran the analyses with various exclusion criteria, such as excluding participants who tapped recommendations more than 90% of the time in a single trial. However, none of these exclusions changed the overall results, so we chose not to exclude any data in the final analysis.

5.2 Participant Experience

Participants often remarked on whether the "shortcuts" were accurate or if they saved them time or effort. Many comments were favorable: "Helped me save a lot of time." However, some participants noted that the benefit of the shortcuts came at the cost of distraction: "It was very pleasant as I did not have to write out all the words. But I think I didn't save much time using it, as I was constantly only looking whether the word I was wanting to write appeared in the box." and "It was nice to have them, but not worth the trouble."

Several participants commented about a mismatch between the sentiment of the recommendations and what they were trying to write, and one participant said "At times I felt like the predictions were guiding my writing."

Some participants noted that the recommendations tended to be generic: "the responses lacked specificity and were difficult to incorporate"; "They definitely make my writing more generic, but I don't mind that." Since the recommendations were chosen to be those that were the most likely, it is unsurprising that they should be perceived as generic. Future work could investigate how to offer recommendations that help writers be more specific.

An error caused our Likert-scale surveys not to be administered, so we quantified participant experiences with the recommendations by coding the open-ended responses that most participants gave after each trial. For each response, blind to condition, one of the authors rated whether it included any favorable remarks about the recommendations (on a scale of 0=none, 1=possible, 2=clear positive) and separately whether it included any unfavorable remarks (same 0–2 scale). For this rating process, only comments about the content of the recommendations were considered; other kinds of comments (e.g., responsiveness, lack of autocorrect, or the word count target) were ignored. We excluded the 5 participants who gave no intelligible comments for one or more trials, leaving 33 participants. Each participant used each condition twice, so we summed the participant's ratings of favorable comments and of unfavorable comments for each condition. This procedure resulted in four numbers for each participant: SKEW-POS-favorable, SKEW-POS-unfavorable, SKEW-NEG-favorable, and SKEW-NEG-unfavorable.

[8] As expected, the number of stars given was a highly significant predictor of positive content ($F_{4,136.3} = 48.1$, $p < .0001$); block index was not significant ($F_{3,106.4} = 1.91$, n.s.).

[9] This analysis treated the difference in star rating as a continuous variable; ordinal regression gave the same conclusion.

A Wilcoxon signed-rank test showed that participants left more favorable comments after writing in the SKEW-POS condition than in the SKEW-NEG condition (Z=95, p=.0008). The average difference in ratings between favorable comments about SKEW-POS and favorable comments about SKEW-NEG (SKEW-POS-favorable - SKEW-NEG-favorable) was 0.34. However, the difference in negative comments was much less pronounced for unfavorable comments: participants left marginally more unfavorable comments after writing in SKEW-NEG than in SKEW-POS (mean of SKEW-POS-unfavorable - SKEW-POS-unfavorable was -0.14). The difference fails to reach statistical significance after accounting for multiple comparisons (Z=159, p=0.029).

6 Discussion

Our results supported our primary hypothesis: writers given positive recommendations included more positive content. This finding suggests that positively skewed recommendations cause writers to intensify their positive sentiment: if they would have written a mildly positive sentence without recommendations, they instead write a clearly positive sentence when given positive recommendations.

We did not find a corresponding effect of negatively skewed recommendations, but this could be due to the very bias we are studying: since the recommender system we were manipulating was biased positive, our manipulations in the SKEW-NEG condition successfully reduced the positive bias, but the system still tended to present positive recommendations more often than negative ones. Reaching a definitive conclusion about the nature of truly negative recommendations requires additional study with a more sophisticated text generation approach.

We find it particularly concerning that participants gave more favorable comments to the SKEW-POS system. While some participants were able to critically reflect on the system's behavior and realize that it could be biasing their writing, many participants seemed to prefer to write with the system that biased their writing to be more positive.

6.1 Limitations and Threats to Validity

Since this experiment had participants write in artificial circumstances, generalizations to natural conditions must be drawn carefully. The strongest threat to the external validity of our findings is that participants behaved in a way that would "please the experimenter," a kind of participant response bias [12]. Although we used instructions and system features to attempt to focus participants' attention on using the recommendations only for efficiency, some participants may have felt pressured to use the recommendations more overall. For example, some participants may have felt that the experimenters wanted them to write in a way that allowed them to use the recommended phrases more.

Two aspects of the experiment design may have given participants clues that sentiment valence was important to us. First, we asked for experiences that differed in sentiment valence (though we used the language "above-average experience" and "below-average experience"). Second, we asked for the perceived sentiment of the recommendations after each trial (though among other survey questions). Comments in the closing survey suggest that at least one participant realized that sentiment was interesting to us. Future work should confirm if the results we present still hold in an experimental setting where sentiment is less salient.

7 Conclusion

Rapid advances in machine learning are increasing the range of tasks for which intelligent systems can make predictions as well as the accuracy of those predictions. As predictive models become more deeply integrated into the systems we build, we must consider how not just the *presence* but also the *content* of those predictions affects the people who use those systems to create and express themselves.

Prior research has found that biased datasets can lead to biased behavior of intelligent systems. We add a third link to this chain: those biased outputs can cause biased *human* behavior in the people who are using those systems.

This effect has ethical implications. If the systems we use encourage us to create certain kinds of artifacts rather than others, what autonomy are we ceding in exchange for efficiency? If it becomes extremely easy to write something in favor of a government or corporation but laborious to write something critical, is our speech really free?

Future work may investigate how systems may be designed to have useful intentional biases. For example, biases towards a kind of language that is stereotypical in a domain can help those unfamiliar with that domain (or second-language learners) write in a more stylistically appropriate way. Systems could make recommendations that support members of minority groups in their goals of how much and to whom they reveal markers of their group membership [30]. Biases towards neutral, negative, or more factual review text, if implemented in a socially and technically thoughtful way, may help *reduce* the positive bias of online review data. And perhaps writers on opposing sides in a debate could receive writing assistance that helps them engage with the opposing side or ground their arguments in generally accepted facts.

Our results suggest that intelligent technology does not simply accelerate our work; it shapes what we create. The data used to train these systems shapes the behavior of those systems; our findings suggest that the training data shape the behavior of the people that use those systems as well.

Online Appendix Code to replicate these experiments is available at `https://github.com/kcarnold/sentiment-slant-gi18/`.

Acknowledgments

Bernd Huber, Martin Segado, and the reviewers provided valuable feedback on the manuscript. Adam Kalai and J. Nathan Matias provided conceptual inspiration and guidance. This work was supported in part by a grant from Draper.

References

[1] K. C. Arnold, K. Z. Gajos, and A. T. Kalai. On Suggesting Phrases vs. Predicting Words for Mobile Text Composition. *Proc. 29th Annu. Symp. User Interface Softw. Technol. - UIST '16*, pp. 603–608, 2016. doi: 10.1145/2984511.2984584

[2] S. Barocas and A. Selbst. Big Data's Disparate Impact. *Calif. Law Rev.*, 104(1):671–729, 2016. doi: 10.15779/Z38BG31

[3] L. Benedetti, H. Winnemöller, M. Corsini, and R. Scopigno. Painting with Bob: Assisted Creativity for Novices. In *Proc. 27th Annu. ACM Symp. User interface Softw. Technol. - UIST '14*, pp. 419–428. ACM Press, New York, New York, USA, 2014. doi: 10.1145/2642918.2647415

[4] R. Berrios, P. Totterdell, and S. Kellett. Eliciting mixed emotions: A meta-analysis comparing models, types and measures. *Front. Psychol.*, 6(MAR):1–15, 2015. doi: 10.3389/fpsyg.2015.00428

[5] X. Bi, T. Ouyang, and S. Zhai. Both complete and correct? Multi-Objective Optimization of Touchscreen Keyboard. In *Proc. 32nd Annu. ACM Conf. Hum. factors Comput. Syst. - CHI '14*, pp. 2297–2306. ACM Press, New York, New York, USA, 2014. doi: 10.1145/2556288.2557414

[6] S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python*, vol. 43. 2009. doi: 10.1097/00004770-200204000-00018

[7] T. Bolukbasi, K.-W. Chang, J. Zou, V. Saligrama, and A. Kalai. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. (Nips):1–9, 2016.

[8] A. Caliskan, J. J. Bryson, and A. Narayanan. Semantics derived automatically from language corpora contain human-like biases. 186(April):183–186, 2016. doi: 10.1126/science.aal4230

[9] M.-h. Chen, S.-t. Huang, H.-t. Hsieh, T.-h. Kao, and J. S. Chang. FLOW : A First-Language-Oriented Writing Assistant System. *Acl 2012*, (July):157–162, 2012.

[10] X. Dai, Y. Liu, X. Wang, and B. Liu. WINGS : Writing with Intelligent Guidance and Suggestions. In *ACL*, pp. 25–30, 2014.

[11] M. Dehghani, S. Rothe, E. Alfonseca, and P. Fleury. Learning to Attend, Copy, and Generate for Session-Based Query Suggestion. In *CIKM*, 2017.

[12] N. Dell, V. Vaidyanathan, I. Medhi, E. Cutrell, and W. Thies. "Yours is better!": Participant response bias in HCI. *Proc. 2012 ACM Annu. Conf. Hum. Factors Comput. Syst. - CHI '12*, pp. 1321–1330, 2012. doi: 10.1145/2207676.2208589

[13] L. Findlater and J. McGrenere. Beyond performance: Feature awareness in personalized interfaces. *Int. J. Hum. Comput. Stud.*, 68(3):121–137, 2010. doi: 10.1016/j.ijhcs.2009.10.002

[14] K. Heafield, I. Pouzyrevsky, J. H. Clark, and P. Koehn. Scalable Modified Kneser-Ney Language Model Estimation. *Proc. 51st Annu. Meet. Assoc. Compu- tational Linguist. (Volume 2 Short Pap.*, pp. 690–696, 2013.

[15] K. Hosanagar, D. Fleder, D. Lee, and A. Buja. Will the Global Village Fracture Into Tribes? Recommender Systems and Their Effects on Consumer Fragmentation. *Manage. Sci.*, 60(4):805–823, 2014. doi: 10.1287/mnsc.2013.1808

[16] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing. Toward Controlled Generation of Text. In *ICML*, mar 2017.

[17] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu. Exploring the Limits of Language Modeling. *arXiv1602.02410 [cs]*, 2016.

[18] L. Kaiser, A. N. Gomez, N. Shazeer, A. Vaswani, N. Parmar, L. Jones, and J. Uszkoreit. One Model To Learn Them All. 2017. doi: 10.1007/s11263-015-0816-y

[19] A. Kannan, K. Kurach, S. Ravi, T. Kaufmann, A. Tomkins, B. Miklos, G. Corrado, L. Lukacs, M. Ganea, P. Young, and V. Ramavajjala. Smart Reply: Automated Response Suggestion for Email. In *KDD*, 2016. doi: 10.475/123

[20] P. O. Kristensson and K. Vertanen. The Inviscid Text Entry Rate and its Application as a Grand Goal for Mobile Text Entry. *Proc. 16th Int. Conf. Human-computer Interact. with Mob. devices Serv. - MobileHCI '14*, pp. 335–338, 2014. doi: 10.1145/2628363.2628405

[21] J. H. Lau, A. Clark, and S. Lappin. Grammaticality, Acceptability, and Probability: A Probabilistic View of Linguistic Knowledge. *Cogn. Sci.*, pp. 1–40, 2016. doi: 10.1111/cogs.12414

[22] B. Lee, S. Srivastava, R. Kumar, R. Brafman, and S. R. Klemmer. Designing with interactive example galleries. *Proc. 28th Int. Conf. Hum. factors Comput. Syst. - CHI '10*, p. 2257, 2010. doi: 10.1145/1753326.1753667

[23] X. Liang, Z. Hu, H. Zhang, C. Gan, and E. P. Xing. Recurrent Topic-Transition GAN for Visual Paragraph Generation. 2017.

[24] Z. C. Lipton, S. Vikram, and J. McAuley. Generative Concatenative Nets Jointly Learn to Write and Classify Reviews. 2015.

[25] R. C. Martin, J. E. Crowther, M. Knight, F. P. Tamborello II, and C.-L. Yang. Planning in sentence production: Evidence for the phrase as a default planning scope. *Cognition*, 116(2):177–192, aug 2010. doi: 10.1016/j.cognition.2010.04.010

[26] A. Moore. Introducing Boomerang Respondable: Personal AI Assistant for Writing Better Emails, 2016.

[27] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *Proc. 43rd Annu. Meet. Assoc. Comput. Linguist. - ACL '05*, (1):115–124, 2005. doi: 10.3115/1219840.1219855

[28] P. Quinn and S. Zhai. A Cost-Benefit Study of Text Entry Suggestion Interaction. *Proc. 2016 CHI Conf. Hum. Factors Comput. Syst.*, pp. 83–88, 2016. doi: 10.1145/2858036.2858305

[29] A. Radford, R. Jozefowicz, and I. Sutskever. Learning to Generate Reviews and Discovering Sentiment. 2017.

[30] S. Reddy and K. Knight. Obfuscating Gender in Social Media Writing. *Proc. ACL Work. Comput. Soc. Sci.*, pp. 17–26, 2016.

[31] M. Torrance. Understanding Planning in Text Production. In *Handb. Writ. Res.*, chap. 5, pp. 72–87. Guilford Press, New York, NY, 2 ed., 2015.

[32] M. Torrance, R. Johansson, V. Johansson, and Å. Wengelin. Reading during the composition of multi-sentence texts: an eye-movement study. *Psychol. Res.*, 80(5):729–743, 2016. doi: 10.1007/s00426-015-0683-8

[33] O. Vinyals and Q. Le. A Neural Conversational Model. 37, 2015.

[34] S. Waldron, C. Wood, and N. Kemp. Use of predictive text in text messaging over the course of a year and its relationship with spelling, orthographic processing and grammar. *J. Res. Read.*, 00(00):1–19, 2016. doi: 10.1111/1467-9817.12073

[35] S. Wang and C. Manning. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. *Proc. 50th Annu. Meet. Assoc. Comput. Linguist.*, (July):90–94, 2012.

[36] J. Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 9909 LNCS:597–613, 2016. doi: 10.1007/978-3-319-46454-1_36