Learning to Relate Literal and Sentimental Descriptions of Visual Properties

Mark Yatskar
Computer Science & Engineering
University of Washington
Seattle, WA
my89@cs.washington.edu

Svitlana Volkova
Center for Language and Speech Processing
Johns Hopkins University
Baltimore, MD
svitlana@jhu.edu

Asli Celikyilmaz
Microsoft
Mountain View, CA
asli@ieee.org

Bill Dolan
NLP Group
Microsoft Research
Redmond, WA
billdol@microsoft.edu

Luke Zettlemoyer
Computer Science & Engineering
University of Washington
Seattle, WA
lsz@cs.washington.edu

Abstract

Language can describe our visual world at many levels, including not only what is literally there but also the sentiment that it invokes. In this paper, we study visual language, both literal and sentimental, that describes the overall appearance and style of virtual characters. Sentimental properties, including labels such as "youthful" or "country western," must be inferred from descriptions of the more literal properties, such as facial features and clothing selection. We present a new dataset, collected to describe Xbox avatars, as well as models for learning the relationships between these avatars and their literal and sentimental descriptions. In a series of experiments, we demonstrate that such learned models can be used for a range of tasks, including predicting sentimental words and using them to rank and build avatars. Together, these results demonstrate that sentimental language provides a concise (though noisy) means of specifying low-level visual properties.

1 Introduction

Language can describe varied aspects of our visual world, including not only what is literally there but also the social, cultural, and emotional sentiment it invokes. Recently, there has been a growing effort to study *literal* language that describes directly observable properties, such as object color, shape, or category (Farhadi et al., 2009; Mitchell et al., 2010; Matuszek et al., 2012). Here, we add a focus on *sentimental* visual language, which compactly describes more subjective properties such as if a person looks determined, if a resume looks professional, or if a restaurant looks romantic. Such models enable many new applications, such as text editors that automatically select properties including font, color, or text alignment to best match high level descriptions such as "professional" or "artistic."

This is a light tan young man with short and trim haircut. He has straight eyebrows and large brown eyes. He has a neat and trim appearance.

State of mind: angry, upset, determined. Likes: country western, rodeo. Occupation: cowboy, wrangler, horse trainer. Overall: youthful, cowboy.

Figure 1: (A) Literal avatar descriptions and (B) sentimental descriptions of four avatar properties, including possible occupations and interests.

In this paper, we study visual language, both literal and sentimental, that describes the overall appearance and style of virtual characters, like those in Figure 1. We use literal language as feature norms, a tool used for studying semantic information in cognitive science (Mcrae et al., 2005). Literal words, such "black" or "hat," are annotated for objects to indicate how people perceive visual properties. Such feature norms provide our gold-standard visual detectors, and allow us to focus on learning to model sentimental language, such as "youthful" or "goth."

We introduce a new corpus of descriptions of Xbox avatars created by actual gamers. Each avatar is specified by 19 attributes, including clothing and body type, allowing for more than 10^{20} possibilities. Using Amazon Mechanical Turk,[1] we collected literal and sentimental descriptions of complete avatars and many of their component parts, such as the cowboy hat in Figure 1(B). In all, there are over 100K descriptions. To demonstrate potential for learning, we also report an A/B test which shows that native speakers can use sentimental descriptions to distinguish the labeled avatars from random distractors. This new data will enable study of the relationships between the co-occurring literal and sentimental text in a rich visual setting.[2]

We describe models for three tasks: (i) classifying when words match avatars, (ii) ranking avatars given a description, and (iii) constructing avatars to match a description. Each model includes literal part descriptions as feature norms, enabling us to learn which literal and sentinel word pairs best predict complete avatars.

Experiments demonstrate the potential for jointly modeling literal and sentimental visual descriptions on our new dataset. The approach outperforms several baselines and learns varied relationships between the sentimental and literal descriptions. For example, in one experiment "nerdy student" is predictive of an avatar with features indicating its shirt is "plaid" and glasses are "large" and faces that are not "bearded." We also show that individual sentimental words can be predicted but that multiple avatars can match a single sentimental description. Finally, we use our model to build complete avatars and show that we can accurately predict the sentimental terms annotators ascribe to them.

[1] www.mturk.com

[2] Data available at http://homes.cs.washington.edu/~my89/avatar.

2 Related Work

To the best of our knowledge, our focus on learning to understand visual sentiment descriptions is novel. However, visual sentiment has been studied from other perspectives. Jrgensen (1998) provides examples which show that visual descriptions communicate social status and story information in addition to literal object and properties. Tousch et al. (2012) draw the distinction between "of-ness" (objective and concrete) and "about-ness" (subjective and abstract) in image retrieval, and observe that many image queries are abstract (for example, images about freedom). Finally, in descriptions of people undergoing emotional distress, Fussell and Moss (1998) show that literal descriptions co-occur frequently with sentimental ones.

There has been significant work on more literal aspects of grounded language understanding, both visual and non-visual. The WordsEye project (Coyne and Sproat, 2001) generates 3D scenes from literal paragraph-length descriptions. Generating literal textual descriptions of visual scenes has also been studied, including both captions (Kulkarni et al., 2011; Yang et al., 2011; Feng and Lapata, 2010) and descriptions (Farhadi et al., 2010). Furthermore, Chen and Dolan (2011) collected literal descriptions of videos with the goal of learning paraphrases while Zitnick and Parikh (2013) describe a corpus of descriptions for clip art that supports the discovery of semantic elements of visual scenes.

There has also been significant recent work on automatically recovering visual attributes, both absolute (Farhadi et al., 2009) and relative (Kovashka et al., 2012), a challenge that we avoid having to solve with our use of feature norms (Mcrae et al., 2005).

Grounded language understanding has also received significant attention, where the goal is to learn to understand situated non-visual language use. For example, there has been work on learning to execute instructions (Branavan et al., 2009; Chen and Mooney, 2011; Artzi and Zettlemoyer, 2013), provide sports commentary (Chen et al., 2010), understand high level strategy guides to improve game

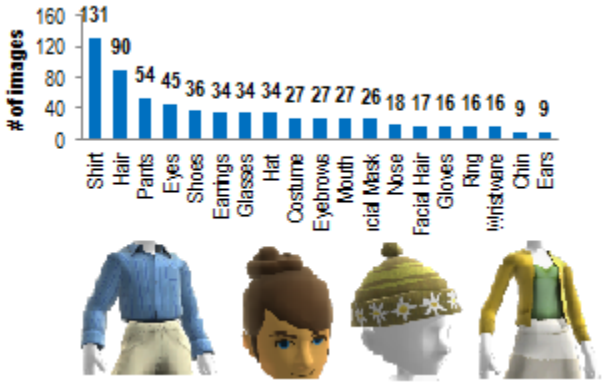

Figure 2: The number of assets per category and example images from the *hair, shirt* and *hat* categories.

play (Branavan et al., 2011; Eisenstein et al., 2009), and understand referring expression (Matuszek et al., 2012).

Finally, our work is similar in spirit to sentiment analysis (Pang et al., 2002), emotion detection from images and speech (Zeng et al., 2009), and metaphor understanding (Shutova, 2010a; Shutova, 2010b). However, we focus on more general visual context.

3 Data Collection

We gathered a large number of natural language descriptions from Mechanical Turk (MTurk). They include: (1) literal descriptions of specific facial features, clothing or accessories and (2) high level subjective descriptions of human-generated avatars.[3]

Literal Descriptions We showed annotators a single image of clothing, a facial feature or an accessory and asked them to produce short descriptions. Figure 2 shows the distribution over object types. We restricted descriptions to be between 3 and 15 words. In all, we collected 33.2K descriptions and had on average 7 words per descriptions. The example annotations with highlighted overlapping patterns are in Table 1.

Sentimental Descriptions We also collected 1913 gamer-created avatars from the web. The avatars were filtered to contain only items from the set of 665 for which we gathered literal descriptions. The gender distribution is 95% male.

[3](2) also has phrases describing emotional reactions. We also collected (3) multilingual literal, (4) relative literal and (5) comprehensive full-body descriptions. We do not use this data, but it will be included in the public release.

LITERAL DESCRIPTIONS
full-sleeved executive blue shirt
blue , long-sleeved button-up shirt
mens blue button dress shirt with dark blue stripes
multi-blue striped long-sleeve button-up dress shirt with cuffs and breast pocket

Table 1: Literal descriptions of shirt in Figure 2.

To gather high level sentimental descriptions, annotators were presented with an image of an avatar and asked to list phrases in response to the follow different aspects:

- State of mind of the avatar.
- Things the avatar might care about.
- What the avatar might do for a living.
- Overall appearance of the avatar.

6144 unique vocabulary items occurred in these descriptions, but only 1179 occurred more than 10 times. Figure 1 (B) shows an avatar and its corresponding sentimental descriptions.

Quality Control All annotations in our dataset are produced by non-expert annotators. We relied on manual spot checks to limit poor annotations. Over time, we developed a trusted crowd of annotators who produced only high quality annotations during the earliest stage of data collection.

4 Feasibility

Our hypothesis is that sentimental language does not uniquely identify an avatar, but instead summarizes or otherwise describes its overall look. In general, there is a trade off between concise and precise descriptions. For example, given a single word you might be able to generally describe the overall look of an avatar, but a long, detailed, literal description would be required to completely specify their appearance.

To demonstrate that the sentimental descriptions we collected are precise enough to be predictive of appearance, we conducted an experiment that prompts people to judge when avatars match descriptions. We created an A/B test where we show English speakers two avatars and one sentimental description. They were asked to select which avatar is better matched by the description and how difficult they felt, on a scale from 1 to 4, it was to judge. For 100 randomly selected descriptions, we

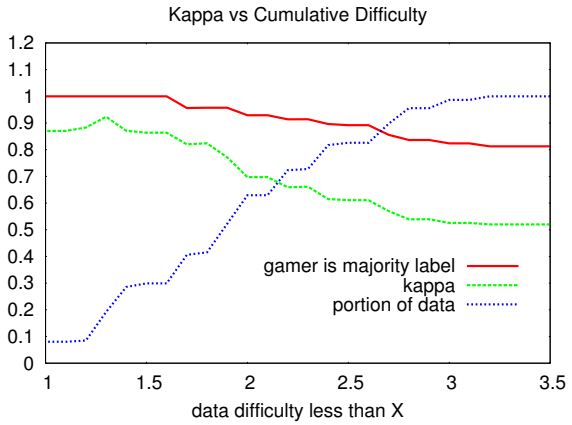

Figure 3: Judged task difficulty versus agreement, gamer avatar preference, and percentage of data covered. The difficulty axis is cumulative.

asked 5 raters to compare the gamer avatars to randomly generated ones (where each asset is selected independently according to a uniform distribution). Figure 3 shows a plot of Kappa and the percent of the time a majority of the raters selected the gamer avatar. The easiest 20% of the data pairs had the strongest agreement, with kappa=.92, and two thirds of the data has kappa = .70. While agreement falls off to .52 for the full data set, the gamer avatar remains the majority judgment 81% of the time.

The fact that random avatars are sometimes preferred indicates that it can be difficult to judge sentimental descriptions. Consider the avatars in Figure 4. Neither conforms to a clear sentimental description based on the questions we asked. The right one is described with conflicting words and the words describing the left one are very general (like "dumb"). This corresponds to our intuition that while many avatars can be succinctly summarized with our questions, some would be more easily described using literal language.

5 Tasks and Evaluation

We formulate three tasks to study the feasibility of learning the relationship between sentimental and literal descriptions. In this section, we first define the space of possible avatars, followed by the tasks.

Avatars Figure 5 summarizes the notation we will develop to describe the data. An avatar is defined by a 19 dimensional vector $\vec{a}$ where each position is an

State of mind: playful, happy; Likes: sex Occupation: hobo Overall: dumb

State of mind: content, humble, satisfied, peaceful, relaxed, calm. Likes: fashion, friends, money, cars, music, education. Occupation: teacher, singer, actor, performer, dancer, computer engineer. Overall: nerdy, cool, smart, comfy, easygoing, reserved

Figure 4: Avatars rated as difficult.

index into a list of possible items $\vec{i}$. Each dimension represents a position on the avatar, for example, *hat* or *nose*. Each possible item is called an asset and is associated with a set of positions it can fill. Most assets take up exactly one position, while there are a few cases where assets take multiple positions.[4] An avatar $\vec{a}$ is valid if all of its mandatory positions are filled, and no two assets conflict on a position. Mandatory positions include hair, eyes, ears, eyebrows, nose, mouth, chin, shirt, pants, and shoes. All other positions are optional. We refer to this set of valid $\vec{a}$ as A. Practically speaking, if an avatar is not valid, it cannot be reliably rendered graphically.

Each item i is associated with the literal descriptions $\vec{d_i} \in D$ where D is the set of literal descriptions. Furthermore, every avatar $\vec{a}$ is associated a list of sentimental query words $\vec{q}$, describing subjective aspects of an avatar.[5]

Sentimental Word Prediction We first study individual words. The word prediction task is to decide whether a given avatar can be described with a

[4]For example, long sleeve shirts cover up watches, so they take up both shirt and wristwear positions. Costumes tend to span many more positions, for example there a suit that takes up shirt, pants, wristwear and shoes positions.

[5]We do not distinguish which prompt (e.g., "state of mind" or "occupation") a word in $\vec{q}$ came from, although the vocabularies are relatively disjoint.

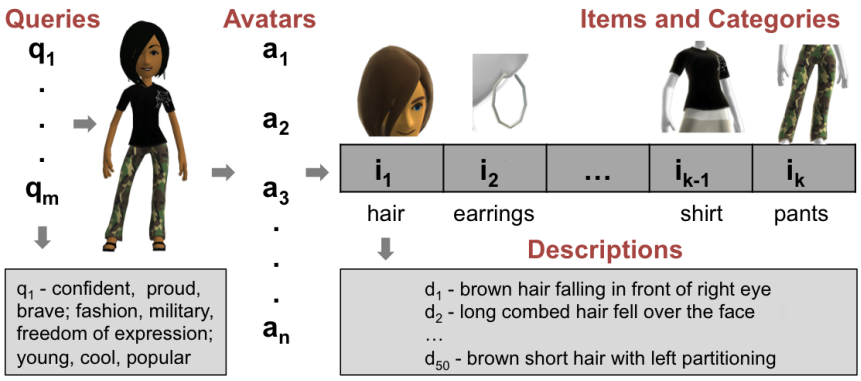

Figure 5: Avatars, queries, items, literal descriptions.

particular sentimental word q^*. We evaluate performance with F-score.

Avatar Ranking We also consider an avatar retrieval task, where the goal is to rank the set of avatars in our data, $\cup_{j=1...n}\vec{a_j}$, according to which one best matches a sentimental description, $\vec{q_i}$. As an automated evaluation, we report the average percentile position assigned to the true $\vec{a_i}$ for each example. However, in general, many different avatars can match each $\vec{q_i}$, an interesting phenomena we will further study with human evaluation.

Avatar Generation Finally, we consider the problem of generating novel, previously unseen avatars, by selecting a set of items that best embody some sentimental description. As with ranking, we aim to construct the avatar $\vec{a_i}$ that matches each sentimental description $\vec{q_i}$. We evaluate by considering the item overlap between $\vec{a_i}$ and the output avatar $\vec{a^*}$, discounting for empty positions:[6]

$$f = \frac{\sum_{j=1}^{|\vec{a^*}|} I(\vec{a_j^*} = \vec{a_{ij}})}{\max(numparts(\vec{a^*}), numparts(\vec{a_i}))}, \quad (1)$$

where $numparts$ returns the number of non-empty avatar positions. The score is a conservative measure because some items are significantly more visually salient than others. For instance, shirts and pants occupy a large portion of the physical realization of the avatar, while rings are small and virtually unnoticeable. We additionally perform a human evaluation in Section 8 to better understand these challenges.

[6]Optional items are infrequently used. Therefore not predicting them at all offers a strong baseline. Yet doing this demonstrates nothing about an algorithm's ability to predict items which contribute to the sentimental qualities of an avatar.

6 Methods

We present two different models: one that considers words in isolation and another that jointly models the query words. This section defines the models and how we learn them.

6.1 Independent Sentimental Word Model

The independent word model (S-Independent) assumes that each word independently describes the avatar. We construct a separate linear model for each word in the vocabulary.

To train these model, we transform the data to form a binary classification problem for each word, where the positive data includes all avatars the word was seen with, $(q, \vec{a_i}, 1)$ for all i and $q \in \vec{q_i}$, and the rest are negative, $(q, \vec{a_i}, 0)$ for all i and $q \notin \vec{q_i}$.

We use the following features:

- an indicator feature for the cross product of a sentiment query word q, a literal description word $w \in D$, and the avatar position index j (for example, $q =$ "angry" with $w =$ "pointy" and $j =$ eyebrows):

$$I(q \in \vec{q_i}, w \in \vec{d_{a_{ij}}}, j)$$

- a bias feature for keeping a position empty:

$$I(q \in \vec{q_i}, a_{ij} = empty, j)$$

These features will allow the model to capture correlations between our feature norms which provide descriptions of visual attributes, like black, and sentimental words, like gothic.

S-Independent is used for both word prediction and ranking. For prediction, we train a linear model using averaged binary perceptron. For ranking, we try to rank all positive instances above negative instances. We use an averaged structured perceptron to train the ranker (Collins, 2002). To rank with respect to an entire query $\vec{q_i}$, we sum the scores of each word $q \in \vec{q_i}$.

6.2 Joint Sentimental Model

The second approach (S-Joint) jointly models the query words to learn the relationships between literal and sentimental words with score s:

$$s(\vec{a}|\vec{q}, D) = \sum_{i=1}^{|\vec{a}|} \sum_{j=1}^{|\vec{q}|} \theta^T f(\vec{a_i}, \vec{q_j}, \vec{d_{a_i}})$$

Where every word in the query has a separate factor and every position is treated independently subject to the constraint that $\vec{a}$ is valid. The feature function f uses the same features as the word independent model above.

This model is used for ranking and generation. For ranking, we try to rank the avatar a_i for query q_i above all other avatars in the candidate set. For generation, we try to score a_i above all other valid avatars given the query q_i. In both cases, we train with averaged structured perceptron (Collins, 2002) on the original data, containing query, avatar pairs $(\vec{q_i}, \vec{a_i})$.

7 Experimental Setup

Random Baseline For the ranking and avatar generation tasks, we report random baselines. For ranking, we randomly order the avatars. In the generation case, we select an item randomly for every position. This baseline does not generate optional assets because they are rare in the real data.

Sentimental-Literal Overlap (SL-Overlap) We also report a baseline that measures the overlap between words in the sentiment query $\vec{q_i}$ and words in the literal asset descriptions D. In generation, for each position in the avatar, $\vec{a_i}$, SL-Overlap selects the item whose literal description has the most words in common with $\vec{q_i}$. If no item had overlap with the query, we backoff to a random choice. In the case of ranking, it orders avatars by the sum over every position of the number of words in common between the literal description and the query, $\vec{q_i}$. This baseline tests the degree to which literal and sentimental descriptions overlap lexically.

Word	F-Score	Precision	Recall	N
happi	0.84	0.89	0.78	149
student	0.78	0.82	0.74	129
friend	0.76	0.84	0.70	153
music	0.74	0.89	0.63	148
confid	0.74	0.82	0.76	157
sport	0.69	0.62	0.76	76
casual	0.63	0.6	0.67	84
youth	0.6	0.57	0.64	88
waitress	0.59	0.42	1	5
smart	0.57	0.54	0.6	88
fashion	0.54	0.54	0.54	70
monei	0.54	0.52	0.56	76
cool	0.54	0.52	0.56	84
relax	0.53	0.52	0.56	90
game	0.51	0.44	0.62	61
musician	0.51	0.44	0.61	66
parti	0.51	0.43	0.62	58
content	0.5	0.47	0.53	75
friendli	0.49	0.42	0.6	56
smooth	0.49	0.4	0.63	57

Table 2: Top 20 words (stemmed) for classification. N is the number of occurances in the test set.

Feature Generation For all models that use lexical features, we limited the number of words. 6144 unique vocabulary items occur in the query set, and 3524 in the literal description set. There are over 400 million entries in the full set of features that include the cross product of these sets with all possible avatar positions, as described in Section 6. Since this would present a challenge for learning, we prune in two ways. We stem all words with a Porter stemmer. We also filter out all features which do not occur at least 10 times in our training set. The final model has approximately 700k features.

8 Results

We present results for the tasks described in Section 5 with the appropriate models from Section 6.

8.1 Word Prediction Results

The goal of our first experiment is to study when individual sentiment words can be accurately predicted. We computed sentimental word classification accuracy for 1179 word classes with 10 or more

Algorithm	Percentile Rank
S-joint	77.3
S-independant	73.5
SL-overlap	60.4
Random	48.8

Table 3: Automatic evaluation of ranking. The average percentile that a test avatar was ranked given its sentimental description.

mentions. Table 2 shows the top 20 words ordered by F-score.[7] Many common words can be predicted with relatively high accuracy. Words with strong individual cues like happy (a smiling mouth), and confidence (wide eyes) and nerdi (particular glasses) can be predicted well.

The average F-score among all words was .085. 33.2% of words have an F-score of zero. These zeros include words like: unusual, bland, sarcastic, trust, prepared, limber, healthy and poetry. Some of these words indicate broad classes of avatars (e.g., unusual avatars) and others indicate subtle modifications to looks that without other words are not specific (e.g., a prepared surfer vs. a prepared business man). Furthermore, evaluation was done assuming that when a word is not mentioned, it is should be predicted as negative. This fails to account for the fact that people do not mention everything that's true, but instead make choices about what to mention based on the most relevant qualities. Despite these difficulties, the classification performance shows that we can accurately capture usage patterns for many words.

8.2 Ranking Results

Ranking allows us to test the hypothesis that multiple avatars are valid for a high level description. Furthermore, we consider the differences between S-Joint and S-Independent, showing that jointly modelings all words improves ranking performance.

Automatic Evaluation The results are shown in Table 3. Both S-Independent and S-Joint outperform the SL-overlap baseline. SL-Overlap's poor performance can be attributed to low direct overlap between sentimental words and literal words. S-Joint also outperforms the S-Independent.

[7] Accuracy numbers are inappropriate in this case because the number of negative instances, in most cases, is far larger than the number of positive ones.

Inspection of the parameters shows that S-Joint does better than S-Independent in modeling words that only relate to a subset of body positions. For example, in one case we found that for the word "puzzled" nearly 50% of the weights were on features that related to eyebrows and eyes. This type of specialization was far more pronounced for S-Joint. The joint nature of the learning allows the features for individual words to specialize for specific positions. In contrast, S-Independent must independently predict all parts for every word.

Human Evaluation We report human relevancy judgments for the top-5 returned results from S-Joint. On average, 56.2% were marked to be relevant. This shows that S-Joint is performing better than automatic numbers would indicate, confirming our intuition that there is a one-to-many relationship between a sentimental description and avatars. Sentimental descriptions, while having significant signal, are not exact. These results also indicate that relying on automatic measures of accuracy that assume a single reference avatar underestimates performance. Figure 6 shows the top ranked results returned by S-Joint for a sentimental description where the model performs well.

8.3 Generation Results

Finally we evaluate three models for avatar generation: Random, SL-Overlap and S-Joint using automatic measures and human evaluation.

Automatic Evaluation Table 4 presents results for automatic evaluation. The Random baseline performs badly, on average assigning items correctly to less than 1 position in the generated avatar. The SL-Overlap baseline improves, but still performs quite poorly. The S-Joint model performs significantly better, correctly guessing 2-3 items for each output avatar. However, as we will see in the manual evaluation, many of the non-matching parts it produces are still a good fit for the query.

Human Evaluation As before, there are many reasonable avatars that could match as well as the reference avatars. Therefore, we also evaluated generation with A/B tests, much like in Section 4. Annotators were asked to judge which of two avatars better matched a sentimental description. They

pensive,confrontational; music,socializing; musician,bar tending,club owner; smart,cool.

Figure 6: A sentimental description paired with the highest ranked avatars found by S-Joint.

Model	Overlap
Random	0.041
SL-Overlap	0.049
S-Joint	0.126

Table 4: Automatic generation evaluation results. The item overlap metric is defined in Section 5.

	Kappa	Majority	Random	Sys.
SL-Overlap	0.20	0.25	0.34	0.32
S-Joint	0.52	0.90	0.07	0.81
Gamer	0.52	0.81	0.08	0.77

Table 5: Human evaluation of automatically generated avatars. Majority represents the percentage of time the system output is preferred by a majority of raters. Random and System (Sys.) indicate the percentage of time each was preferred.

could rate System A or System B as better, or report that they were equal or that neither matched the description. We consider two comparisons: SL-Overlap vs. Random and S-Joint vs Random. Five annotators performed each condition, rating 100 examples with randomly ordered avatars.

We report the results for human evaluation including kappa, majority judgments, and a distribution over judgments in Table 5. The SL-Overlap baseline is indistinguishable from a random avatar. This contrasts with the ranking case, where this simple baseline showed improvement, indicating that generation is a much harder problem. Furthermore, agreement is low; people felt the need to make a choice but were not consistent.

We also see in Table 5 that people prefer the S-Joint model outputs to random avatars as often as they prefer gamer to random. While this does not necessarily imply that S-Joint creates gamer-quality avatars, it indicates substantial progress by learning a mapping between literal and sentimental words.

Qualitative Results Table 6 presents the highest and lowest weighted features for different sentimental query words. Figure 7 shows four descriptions that were assigned high quality avatars.

In general, many of the weaker avatars had aspects of the descriptions but lacked such distinctive overall looks. This was especially true when the descriptions contained seemingly contradictory information. For example, one avatar was described as being both nerdy and popular. We generated a look that had aspects of both of these descriptions, including a head that contained both conservative elements (like glasses) and less conservative elements (like crazy hair and earrings). However, the combination would not be described as nerdy or popular, because of difficult to predict global interactions between the co-occurring words and items. This is an important area for future work.

9 Conclusions

We explored how visual language, both literal and sentimental, maps to the overall physical appearance and style of virtual characters. While this paper focused on avatar design, our approach has implications for a broad class of natural language-driven

Ambition; business, fashion, success; salesman; smooth, professional.

Capable, confident, firm; heavy metal, extreme sports, motorcycles; engineer, mechanic, machinist; aggressive, strong, protective.

Stressed, bored, discontent; emo music; works at a record store; goth, dark, drab.

Happy, content, confident, home, career, family, secretary,student, classy,clean,casual

Figure 7: Avatars automatically generated with the S-Joint model.

Sentiment	positive features	negative features
happi	mouth:thick, mouth:smilei, mouth:make, mouth:open	mouth:tight, mouth:emotionless, mouth:brownish, mouth:attract
gothic	shoes:brown, shirt:black, pants:hot, shirt:band	shirt:half, shirt:tight, pants:sexi, hair:brownish
retro	eyebrows:men, eyebrows:large, hair:round, pants:light	eyebrows:beauti, pants:side; eyebrows:trim, pants:cut
beach	pants:yello, pants:half, nose:narrow, pants:white	shirt:brown, shirt:side; shoes:long, pants:jean

Table 6: Most positive and negative features for a word stem. A feature is [position]:[literal word].

dialog scenarios. In many situations, a user may be perfectly able to formulate a high-level description of their intent ("Make my resume look cleaner" "Buy me clothes for a summer wedding," or "Play something more danceable") while having little or no understanding of the complex parameter space that the underlying software must manipulate in order to achieve this result.

We demonstrated that these high-level sentimental specifications can have a strong relationship to literal aspects of a problem space and showed that sentimental language is a concise, yet noisy, way of specifying high level characteristics. Sentimental language is an unexplored avenue for improving natural language systems that operate in situated settings. It has the potential to bridge the gap between lay and expert understandings of a problem domain.

Acknowledgments

This work is partially supported by the DARPA CSSG (N11AP20020) and the NSF (IIS-1115966). The authors would like to thank Chris Brockett, Noelle Sophy, Rico Malvar for helping with collecting and processing the data. We would also like to thank Tom Kwiatkowski and Nicholas FitzGerald and the anonymous reviewers for their helpful comments.

References

Yoav Artzi and Luke Zettlemoyer. 2013. Weakly supervised learning of semantic parsers for mapping instructions to actions. *Transactions of the Association for Computational Linguistics*, 1(1):49–62.

SRK Branavan, H. Chen, L.S. Zettlemoyer, and R. Barzilay. 2009. Reinforcement learning for mapping instructions to actions. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 82–90.

SRK Branavan, David Silver, and Regina Barzilay. 2011. Learning to win by reading manuals in a monte-carlo framework. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 268–277.

David L. Chen and William B. Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 190–200.

D.L. Chen and R.J. Mooney. 2011. Learning to interpret natural language navigation instructions from observa-

tions. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI-2011)*, pages 859–865.

David L. Chen, Joohyun Kim, and Raymond J. Mooney. 2010. Training a multilingual sportscaster: Using perceptual context to learn language. *Journal of Artificial Intelligence Research*, 37:397–435.

Michael Collins. 2002. Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 1–8.

B. Coyne and R. Sproat. 2001. Wordseye: an automatic text-to-scene conversion system. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 487–496.

J. Eisenstein, J. Clarke, D. Goldwasser, and D. Roth. 2009. Reading to learn: Constructing features from semantic abstracts. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 958–967.

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. 2010. Every picture tells a story: generating sentences from images. In *Proceedings of the 11th European conference on Computer Vision*, ECCV'10, pages 15–29.

Yansong Feng and Mirella Lapata. 2010. Topic models for image annotation and text illustration. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 831–839.

Susan R Fussell and Mallie M Moss. 1998. Figurative language in emotional communication. *Social and cognitive approaches to interpersonal communication*, page 113.

Corinne Jrgensen. 1998. Attributes of images in describing tasks. *Information Processing & Management*, 34(23):161 – 174.

Adriana Kovashka, Devi Parikh, and Kristen Grauman. 2012. Whittlesearch: Image search with relative attribute feedback. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2973–2980.

G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A.C. Berg, and T.L. Berg. 2011. Baby talk: Understanding and generating simple image descriptions. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1601–1608.

Cynthia Matuszek, Nicholas FitzGerald, Luke Zettlemoyer, Liefeng Bo, and Dieter Fox. 2012. A Joint Model of Language and Perception for Grounded Attribute Learning. In *Proc. of the 2012 International Conference on Machine Learning*.

Ken Mcrae, George S. Cree, Mark S. Seidenberg, and Chris Mcnorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behavior Research Methods*, 37(4):547–559.

Margaret Mitchell, Kees van Deemter, and Ehud Reiter. 2010. Natural reference to objects in a visual domain. In *Proceedings of the 6th International Natural Language Generation Conference*, INLG '10, pages 95–104.

B. Pang, L. Lee, and S. Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 79–86.

Ekaterina Shutova. 2010a. Automatic metaphor interpretation as a paraphrasing task. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 1029–1037.

Ekaterina Shutova. 2010b. Models of metaphor in nlp. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 688–697.

Anne-Marie Tousch, Stphane Herbin, and Jean-Yves Audibert. 2012. Semantic hierarchies for image annotation: A survey. *Pattern Recognition*, 45(1):333 – 345.

Yezhou Yang, Ching Lik Teo, Hal Daumé III, and Yiannis Aloimonos. 2011. Corpus-guided sentence generation of natural images. In *Empirical Methods in Natural Language Processing*.

Z. Zeng, M. Pantic, G.I. Roisman, and T.S. Huang. 2009. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(1):39–58.

C Lawrence Zitnick and Devi Parikh. 2013. Bringing semantics into focus using visual abstraction. In *Computer Vision and Pattern Recognition (To Appear)*.