

# I Can Has Cheezburger?

## A Nonparanormal Approach to Combining Textual and Visual Information for Predicting and Generating Popular Meme Descriptions

William Yang Wang and Miaomiao Wen  
School of Computer Science  
Carnegie Mellon University  
Pittsburgh, PA 15213

### Abstract

The advent of social media has brought Internet memes, a unique social phenomenon, to the front stage of the Web. Embodied in the form of images with text descriptions, little do we know about the “language of memes”. In this paper, we statistically study the correlations among popular memes and their wordings, and generate meme descriptions from raw images. To do this, we take a multimodal approach—we propose a robust nonparanormal model to learn the stochastic dependencies among the image, the candidate descriptions, and the popular votes. In experiments, we show that combining text and vision helps identifying popular meme descriptions; that our nonparanormal model is able to learn dense and continuous vision features jointly with sparse and discrete text features in a principled manner, outperforming various competitive baselines; that our system can generate meme descriptions using a simple pipeline.

### 1 Introduction

In the past few years, Internet memes become a new, contagious social phenomenon: it all starts with an image with a witty, catchy, or sarcastic sentence, and people circulate it from friends to friends, colleagues to colleagues, and families to families. Eventually, some of them go viral on the Internet.

Meme is not only about the funny picture, the Internet culture, or the emotion that passes along, but also about the richness and uniqueness of its language: it is often highly structured with special written style, and forms interesting and subtle connotations that resonate among the readers. For example, the LOL cat memes (e.g., Figure 1) often



Figure 1: An example of the LOL cat memes.

include superimposed text with broken grammars and/or spellings.

Even though the memes are popular over the Internet, the “language of memes” is still not well-understood: there are no systematic studies on predicting and generating popular Internet memes from the Natural Language Processing (NLP) and Computer Vision (CV) perspectives.

In this paper, we take a multimodal approach to predict and generate popular meme descriptions. To do this, we collect a set of original meme images, a list of candidate descriptions, and the corresponding votes. We propose a robust nonparanormal approach (Liu et al., 2009) to model the multimodal stochastic dependencies among images, text, and votes. We then introduce a simple pipeline for generating meme descriptions combining reverse image search and traditional information retrieval approaches. In empirical experiments, we show that our model outperforms strong discriminative baselines by very large margins in the regression/ranking experiments, and that in the generation experiment, the nonparanormal outperforms the second-best supervised baseline by 4.35 BLEU points, and obtains a BLEU score improvement of 4.48 over an unsupervised recurrent neural network language model

trained on a large meme corpus that is almost 90 times larger. Our contributions are three-fold:

- We are the first to study the “language of memes” combining NLP, CV, and machine learning techniques, and show that combining the visual and textual signals helps identifying popular meme descriptions;
- Our approach empowers Internet users to select better wordings and generate new memes automatically;
- Our proposed robust nonparanormal model outperforms competitive baselines for predicting and generating popular meme descriptions.

In the next section, we outline related work. In Section 3, we introduce the theory of copula, and our nonparanormal approach. In Section 4, we describe the datasets. We show the prediction and generation results in Section 5 and Section 6. Finally, we conclude in Section 7.

## 2 Related Work

Although the language of Internet memes is a relatively new research topic, our work is broadly related to studies on predicting popular social media messages (Hong et al., 2011; Bakshy et al., 2011; Artzi et al., 2012). Most recently, Tan et al. (2014) study the effect on wordings for Tweets. However, none of the above studies have investigated multi-modal approaches that combine text and vision.

Recently, there has been growing interests in inter-disciplinary research on generating image descriptions. Gupta et al. (2009) have studied the problem of constructing plots from video understanding. The work by Farhadi et al. (2010) is among the first to generate sentences from images. Kulkaarni et al. (2011) use linguistic constraints and a conditional random field model for the task, whereas Mitchell et al. (2012) leverage syntactic information and co-occurrence statistics and Dodge et al. (2012) use a large text corpus and CV algorithms for detecting visual text. With the surge of interests in deep learning techniques in NLP (Socher et al., 2013; Devlin et al., 2014) and CV (Krizhevsky et al., 2012; Oquab et al., 2013), there have been several unreferenced manuscripts on parsing images and generating text descriptions lately (Vinyals et al., 2014; Chen

and Zitnick, 2014; Donahue et al., 2014; Fang et al., 2014; Karpathy and Fei-Fei, 2014) using neural network models. Although the above studies have shown interesting results, our task is arguably more complex than generating text descriptions: in addition to the visual and textual signals, we have to model the popular votes as a third dimension for learning. For example, we cannot simply train a convolutional neural network image parser on billions of images, and use recurrent neural networks to generate texts such as “*There is a white cat sitting next to a laptop.*” for Figure 1. Additionally, since not all images are suitable as meme images, collecting training images is also more challenging in our task.

In contrast to prior work, we take a very different approach: we investigate copula methods (Schweizer and Sklar, 1983; Nelsen, 1999), in particular, the nonparanormals (Liu et al., 2009), for joint modeling of raw images, text descriptions, and popular votes. Copula is a statistical framework for analyzing random variables from Statistics (Liu et al., 2012), and often used in Economics (Chen and Fan, 2006). Only until very recently, researchers from the machine learning and information retrieval communities (Ghahramani et al., 2012; Han et al., 2012; Eickhoff et al., 2013) start to understand the theory and the predictive power of copula models. Wang and Hua (2014) are the first to introduce semi-parametric Gaussian copula (a.k.a. nonparanormals) for text prediction. However, their approach may be prone to overfitting. In this work, we generalize Wang and Hua’s method to jointly model text and vision features with popular votes, while scaling up the model using effective dropout regularization.

## 3 Our Approach

A key challenge for joint modeling of text and vision is that, because textual features are often relatively sparse and discrete, while visual features are typically dense and continuous, it is difficult to model them jointly in a principled way.

To avoid comparing “apple and oranges” in the same probabilistic space, we propose the nonparanormal approach, which extends the Gaussian graphical model by transforming its variables by smooth functions. More specifically, for each dimension of textual and visual features, instead of

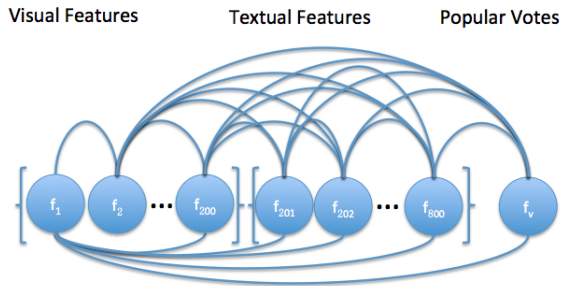


Figure 2: Our nonparanormal method extends Gaussian by transforming each dimension with a smooth function, and jointly models the stochastic dependencies among textual and visual features, as well as the popular votes by the crowd.

using raw counts or histograms, we first use *probability integral transform* to generate empirical cumulative density functions (ECDF): now instead of the probability density function (PDF) space, we are working in the ECDF space where the value of each feature is based on the rank, and is strictly restricted between 0 and 1. Then, we use kernel density estimation to smooth out the zeroing features<sup>1</sup>. Finally, now textual and visual features are compatible, and we then build a parametric Gaussian copula model to estimate the pair-wise correlations among the covariate and the dependent variable.

In this section, we first explain the visual and textual features used in this study. Then, we introduce the theory of copula, and describe the robust nonparanormal. Finally, we show a simple pipeline for generating meme descriptions.

### 3.1 Features

**Textual Features** To model the meme descriptions, we take a broad range of textual features into considerations:

- **Lexical Features:** we extract unigrams and bigrams from meme descriptions as surface-level lexical features.
- **Part-of-Speech Features:** to model shallow syntactic cues, we extract lexicalized part-of-speech features using the Stanford part-of-speech tagger (Toutanova et al., 2003).
- **Dependency Triples:** to better understand the deeper syntactic dependencies of keywords in

<sup>1</sup>This is necessary for the normal inversion of the ECDFs, which we will describe in Section 3.2.



Figure 3: An example of the standard SIFT keypoints detected on the “doge” meme.

memes, we have also extracted typed dependency triples (e.g., *subj(I,are)*) using the Malt-Parser (Nivre et al., 2007).

- **Named Entity Features:** after browsing the dataset, we notice that certain names are often mentioned in memes (e.g. “Drake”, “Kenye West”, and “Justin Bieber”), so we utilize the Stanford named entity recognizer (Finkel et al., 2005) to extract lexicalized named entities.
- **Frame-Semantics Features:** SEMAFOR (Das et al., 2010) is a state-of-the-art frame-semantics parser that produces FrameNet-style semantic annotation. We use SEMAFOR to extract frame-level semantic features.

**Visual Features** A key insight on viral memes is that the images producing a shared social signal are typically inter-related in style. For example, LOLcats are an early series of memes involving funny cat photos. Similarly, “Bieber memes” involve modified pictures of Bieber.

Therefore, we hypothesize that, by extracting visual features, it is of crucial importance to capture the entities, objects, and styles as visual words in these inter-related meme images. The popular visual bag-of-words representation (Sivic and Zisserman, 2003) is used to describe images:

1. **PHOW Features Extraction:** unlike text features, SIFT first detects the Harris keypoints from an image, and then describes each keypoint with a vector. An example of the SIFT frames are shown in Figure 3. PHOW (Bosch et al., 2007) is a dense and multi-scale variant of the Scale Invariant Feature Transform (SIFT) descriptors. Using PHOW, we obtain about 20K keypoints for each image.

2. **Elkan K-means Clustering** is the clustering method (Elkan, 2003) that we use to obtain the vocabulary for visual words. Comparing to other variants of K-means, this method quickly constructs the codebook from PHOW keypoints.
3. **Bag-of-Words Histograms** are used to represent each image. We match the PHOW keypoints of each image with the vocabulary that we extract from the previous step, and generate a  $1 \times 200$  sized visual bag-of-words vector.

### 3.2 The Theory of Copula

In the Statistics literature, copula is widely known as a family of distribution function. The idea behind copula theory is that the cumulative distribution function (CDF) of a random vector can be represented in the form of uniform marginal cumulative distribution functions, and a copula that connects these marginal CDFs, which describes the correlations among the input random variables. However, in order to have a valid multivariate distribution function regardless of  $n$ -dimensional covariates, not every function can be used as a copula function. The central idea behind copula, therefore, can be summarized by the Sklar's theorem and the corollary.

**Theorem 1 (Sklar's Theorem (1959))** *Let  $F$  be the joint cumulative distribution function of  $n$  random variables  $X_1, X_2, \dots, X_n$ . Let the corresponding marginal cumulative distribution functions of the random variable be  $F_1(x_1), F_2(x_2), \dots, F_n(x_n)$ . Then, if the marginal functions are continuous, there exists a unique copula  $C$ , such that*

$$F(x_1, \dots, x_n) = C[F_1(x_1), \dots, F_n(x_n)]. \quad (1)$$

Furthermore, if the distributions are continuous, the multivariate dependency structure and the marginals might be separated, and the copula can be considered independent of the marginals (Joe, 1997; Parsa and Klugman, 2011). Therefore, the copula does not have requirements on the marginal distributions, and any arbitrary marginals can be combined and their dependency structure can be modeled using the copula. The inverse of Sklar's Theorem is also true in the following:

**Corollary 1** *If there exists a copula  $C : (0, 1)^n$  and marginal cumulative distribution functions  $F_1(x_1), F_2(x_2), \dots, F_n(x_n)$ , then*

$C[F_1(x_1), \dots, F_n(x_n)]$  defines a multivariate cumulative distribution function.

### 3.3 The Nonparanormal

To model multivariate text and vision variables, we choose the nonparanormal (NPN) as the copula function in this study, which can be explained in the following two parts.

#### The Nonparametric Estimation

Assume we have  $n$  random variables of vision and text features  $X_1, X_2, \dots, X_n$ . The problem is that text features are sparse, so we need to perform nonparametric kernel density estimation to smooth out the distribution of each variable. Let  $f_1, f_2, \dots, f_n$  be the unknown density, we are interested in deriving the shape of these functions. Assume we have  $m$  samples, the kernel density estimator can be defined as:

$$\hat{f}_h(x) = \frac{1}{m} \sum_{i=1}^m K_h(x - x_i) \quad (2)$$

$$= \frac{1}{mh} \sum_{i=1}^m K\left(\frac{x - x_i}{h}\right) \quad (3)$$

Here,  $K(\cdot)$  is the kernel function, where in our case, we use the Box kernel<sup>2</sup>  $K(z)$ :

$$K(z) = \frac{1}{2}, |z| \leq 1, \quad (4)$$

$$= 0, |z| > 1. \quad (5)$$

Comparing to the Gaussian kernel and other kernels, the Box kernel is simple, and computationally inexpensive. The parameter  $h$  is the bandwidth for smoothing<sup>3</sup>.

Now, we can derive the empirical cumulative distribution functions

$$\hat{F}_{X_1}(\hat{f}_1(X_1)), \hat{F}_{X_2}(\hat{f}_2(X_2)), \dots, \hat{F}_{X_n}(\hat{f}_n(X_n))$$

of the smoothed covariates, as well as the dependent variable  $y$  (which is the reciprocal rank of the popular votes of a meme) and its CDF  $\hat{F}_y(\hat{f}(y))$ . The

<sup>2</sup>It is also known as the original Parzen windows (Parzen, 1962).

<sup>3</sup>In our implementation, we use the default  $h$  of the Box kernel in the `ksdensity` function in Matlab.

empirical cumulative distribution functions are defined as:

$$\hat{F}(\nu) = \frac{1}{m} \sum_{i=1}^m \mathbf{I}\{x_i \leq \nu\} \quad (6)$$

where  $\mathbf{I}\{\cdot\}$  is the indicator function, and  $\nu$  indicates the current value that we are evaluating. Note that the above step is also known as *probability integral transform* (Diebold et al., 1997), which allows us to convert any given continuous distribution to random variables having a uniform distribution. This is crucial for text: instead of using the raw counts, we are now working with uniform marginal CDFs, which helps coping with the overfitting issue due to noise and data sparsity. We also use the same procedure to transform the vision features into CDF space to be compatible with text features.

### The Robust Estimation of Copula

Now that we have obtained the marginals, and then the joint distribution can be constructed by applying the copula function that models the stochastic dependencies among marginal CDFs:

$$\begin{aligned} & \hat{F}(\hat{f}_1(X_1), \dots, \hat{f}_1(X_n), \hat{f}(y)) \\ &= C[\hat{F}_{X_1}(\hat{f}_1(X_1)), \dots, \hat{F}_{X_n}(\hat{f}_n(X_n)), \hat{F}_y(\hat{f}_y(y))] \end{aligned} \quad (7)$$

In this work, we apply the parametric Gaussian copula to model the correlations among the text features and the label. Assume  $x_i$  is the smoothed version of random variable  $X_i$ , and  $y$  is the smoothed label, we have:

$$\begin{aligned} & F(x_1, \dots, x_n, y) \\ &= \Phi_{\Sigma} \left( \Phi^{-1}[F_{x_1}(x_1)], \dots, \Phi^{-1}[F_{x_n}(x_n)], \Phi^{-1}[F_y(y)] \right) \end{aligned} \quad (8)$$

where  $\Phi_{\Sigma}$  is the joint cumulative distribution function of a multivariate Gaussian with zero mean and  $\Sigma$  variance.  $\Phi^{-1}$  is the inverse CDF of a standard Gaussian. In this parametric part of the model, the parameter estimation boils down to the problem of learning the covariance matrix  $\Sigma$  of this Gaussian copula. In this work, we perform standard maximum likelihood estimation (MLE) for the  $\Sigma$  matrix, where we follow the details from prior work (Wang and Hua, 2014).

To avoid overfitting, traditionally, one resorts to classic regularization techniques such as Lasso (Tib-

shirani, 1996). While Lasso is widely used, the non-differentiable nature of the  $L_1$  norm often make the objective function difficult to optimize. In this work, we propose dropout training (Hinton et al., 2012) as copula regularization. Dropout was proposed by Hinton et al. as a method to prevent feature co-adaptation in the deep learning framework, but recently studies (Wager et al., 2013) also show that its behaviour is similar to  $L_2$  regularization, and can be approximated efficiently (Wang and Manning, 2013) in many other machine learning tasks. Another advantage of dropout training is that, unlike Lasso, it does not require all the features for training, and training is “embarrassingly” parallelizable.

In Gaussian copula estimation context, we can introduce another dimension  $\ell$ : the number of dropout learners, to extend the  $\Sigma$  into a dropout tensor. Essentially, the task becomes the estimation of

$$\Sigma_1, \Sigma_2, \dots, \Sigma_{\ell}$$

where the input feature space for each dropout component is randomly corrupted by  $(1 - \delta)$  percent of the original dimension. In the inference time, we use geometric mean to average the predictions from each dropout learner, and generate the final prediction. Note that the final  $\Sigma$  matrix has to be symmetric and positive definite, so we apply tiny random Gaussian noise  $\epsilon$  to maintain the property.

### Computational Complexity

One important question regarding the proposed nonparanormal model is the corresponding computational complexity. This boils down to the estimation of the  $\hat{\Sigma}$  matrix (Liu et al., 2012): one only needs to calculate the correlation coefficients of  $n(n - 1)/2$  pairs of random variables. Christensen (2005) shows that sorting and balanced binary trees can be used to calculate the correlation coefficients with complexity of  $O(n \log n)$ . Therefore, the computational complexity of MLE for the proposed model is  $O(n \log n)$ .

### Efficient Approximate Inference

In this prediction task, in order to perform the exact inference of the conditional probability distribution  $p(F_y(y) | F_{x_1}(x_1), \dots, F_{x_n}(x_n))$ , one needs to solve the mean response  $\hat{\mathbf{E}}(F_y(y) | F_{x_1}(x_1), \dots, F_{x_1}(x_1))$  from a joint distribution of high-dimensional Gaussian copula. Unfortunately, the exact inference can be

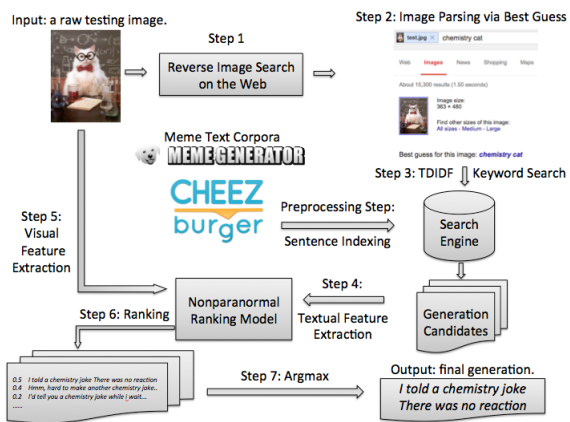


Figure 4: Our pipeline for generating memes from raw images.

intractable in the multivariate case, and approximate inference, such as Markov Chain Monte Carlo sampling (Gelfand and Smith, 1990; Pitt et al., 2006) is often used for posterior inference. In this work, we propose an efficient sampling method to derive  $y$  given the text features — we sample  $F_y(\hat{y})$  s.t. it maximizes the joint high-dimensional Gaussian copula density:

$$\arg \max_{F_y(\hat{y}) \in (0,1)} \frac{1}{\sqrt{\det \Sigma}} \exp \left( -\frac{1}{2} \Delta^T \cdot (\Sigma^{-1} - \mathbf{I}) \cdot \Delta \right) \quad (9)$$

where

$$\Delta = \begin{pmatrix} \Phi^{-1}(F_{x_1}(x_1)) \\ \vdots \\ \Phi^{-1}(F_{x_n}(x_n)) \\ \Phi^{-1}(F_y(y)) \end{pmatrix}$$

This approximate inference scheme using maximum density sampling from the Gaussian copula significantly relaxes the complexity of inference. Finally, to derive  $\hat{y}$ , the last step is to compute the inverse CDF of  $F_y(\hat{y})$ . A detailed description of the inference algorithm can be found in our prior work (Wang and Hua, 2014).

### 3.4 A Simple Meme Generation Pipeline

Now after we train a nonparanormal model for ranking meme descriptions, we show the simple meme generation pipeline in Figure 4.

Given a test image, we disguise as the Internet Explorer, and query Google’s “Search By Image” inverse image search service<sup>4</sup>. By comparing the

<sup>4</sup><http://www.google.com/imgph/>

query image with all possible images with their captions in Google’s database, a “Best Guess” of the keywords in the image is then revealed.

Using the extracted image keywords, we further query a TF-IDF based Lucene<sup>5</sup> meme search engine, which we indexed with a large number of Web-crawled meme descriptions. After we obtain the candidate generations, we then extract all the text and vision features that we described in Section 3.1. Finally, our nonparanormal model ranks all possible candidates, and selects the final generation with the highest posterior.

## 4 Datasets

We collected meme images and text descriptions<sup>6</sup> from two popular meme websites<sup>7</sup>. In the prediction experiment, we use 3,008 image-description pairs for training, and 526 image-description pairs for testing. In the generation experiment, we use 269,473 meme descriptions to index the meme search engine, and 50 randomly selected images for testing. During training, we convert the raw counts of popular votes into reciprocal ranks (e.g., the most popular text descriptions will all have a reciprocal rank of 1, and  $n$ -th popular one will have a score of  $1/n$ ).

## 5 Prediction Experiments

In the first experiment, we compare the proposed NPN with various baselines in a prediction task, since prior literature (Hodosh et al., 2013) also suggests using ranking based evaluation for associating images with text descriptions. Throughout the experiment sections, we set  $\ell = 10$ , and  $\delta = 80$  as the dropout hyperparameters.

### Baselines:

The baselines are standard squared-loss linear regression, linear kernel SVM, and non-linear (Gaussian) kernel SVM. In a recent empirical study (Fernández-Delgado et al., 2014) that evaluates 179 classifiers from 17 families on 121 UCI datasets, the authors find that Gaussian SVM is one of the top performing classifiers. We use the Statistical Toolbox’s linear regression implementation in Matlab, and LibSVM (Chang and Lin, 2011) for

<sup>5</sup><http://lucene.apache.org/>

<sup>6</sup>[http://www.cs.cmu.edu/~yww/data/meme\\_dataset.zip](http://www.cs.cmu.edu/~yww/data/meme_dataset.zip).

<sup>7</sup>[memegenerator.net](http://memegenerator.net) and [cheezburger.com](http://cheezburger.com)

training and testing the SVM models. The hyperparameter  $C$  in linear SVM, and the  $\gamma$  and  $C$  hyperparameters in Gaussian SVM are tuned on the training set using 10-fold cross-validation.

### Evaluation Metrics:

Spearman’s correlation (Hogg and Craig, 1994) and Kendall’s tau (Kendall, 1938) have been widely used in many real-valued prediction (regression) problems in NLP (Albrecht and Hwa, 2007; Yogatama et al., 2011), and here we use them to measure the quality of predicted values  $\hat{y}$  by comparing to the vector of ground truth  $y$ . Kendall’s tau is a nonparametric statistical metric that have shown to be inexpensive, robust, and representation independent (Lapata, 2006). We use paired two-tailed t-test to measure the statistical significance.

### 5.1 Comparison with Various Baselines

The first two figures in Figure 5 show the learning curve of our system, comparing other baselines. We see that when increasing the amount of training data, our approach clearly dominates all other methods by a large margin. Linear and Gaussian SVMs perform similarly, and have good performances with only 25% of the training data, but the improvements are not large when increasing the amount of training data.

In the last two figures in Figure 5, we increase the amount of features, and compare various models. We see that the linear regression model overfits with 600 features, and Gaussian SVM outperforms the linear SVM. We see that our NPN model clearly outperforms all baselines by a big gap, and does not overfit.

### 5.2 Combination of Text and Vision

In Table 1, we systematically compare the contributions of each feature set. First, we see that bigram features clearly improve the performance on top of unigram features. Second, named entities are crucial for further boosting the performance. Third, adding the shallow part-of-speech features does not benefit all models, but the dependency triples are shown to be useful for all methods. Finally, we see that using semantic features helps increasing the performances for most of the cases, and combining text and vision features in our NPN framework doubles the perfor-

Feature Sets	LR	LSVM	GSVM	NPN
Unigrams	0.152	0.158	0.176	<b>0.241*</b>
+ Bigrams	0.163	0.248	0.279	<b>0.318*</b>
+ Named Entities	0.188	0.296	0.312	<b>0.339*</b>
+ Part-of-Speech	0.184	0.318	0.337	<b>0.343</b>
+ Dependency	0.191	0.322	0.348	<b>0.350</b>
+ Semantics	0.183	0.368	<b>0.388</b>	0.367
All Text + Vision	0.413	0.415	0.451	<b>0.754*</b>
Unigrams	0.102	0.105	0.118	<b>0.181*</b>
+ Bigrams	0.115	0.164	0.187	<b>0.237*</b>
+ Named Entities	0.127	0.202	0.213	<b>0.248*</b>
+ Part-of-Speech	0.125	0.218	0.232	<b>0.239</b>
+ Dependency	0.130	0.223	0.242	<b>0.255</b>
+ Semantics	0.124	0.257	<b>0.270</b>	<b>0.270</b>
All Text + Vision	0.284	0.288	0.314	<b>0.580*</b>

Table 1: The Spearman correlation (top table) and Kendall’s  $\tau$  (bottom table) for comparing various text features and combining with vision features. The best results of each row are highlighted in **bold**. \* indicates  $p < .001$  comparing to the second best result.

mance for associating popular votes, meme images, and text descriptions.

### 5.3 The Effects of Dropout Training for Nonparanormals

As we mentioned before, because NPNs model the complex network of random variables, a key issue for training NPN is to prevent the model from overfitting to the training data. So far, none of the prior work have investigated dropout training for regularizing the nonparanormals or even copula in general. To empirical test the effects of dropout training for nonparanormals, in addition to our datasets, we also compare with the unregularized copula from Wang and Hua (2014) on predicting financial risks from earnings calls. Table 2 clearly suggests that dropout training for NPNs significant improves the performances on various datasets.

### 5.4 Qualitative Analysis

Table 3 shows the top ranked text features that are highly correlated with popular votes. We see that the named entity features are useful: Paul Walker, UPS, Bruce Willis, Pencil Guy, Amy Winehouse are recognized as entities in the meme dataset. Dependency triples, as a less-understood feature set, also perform well in this task. For example,  $xcomp(tell,mean)$

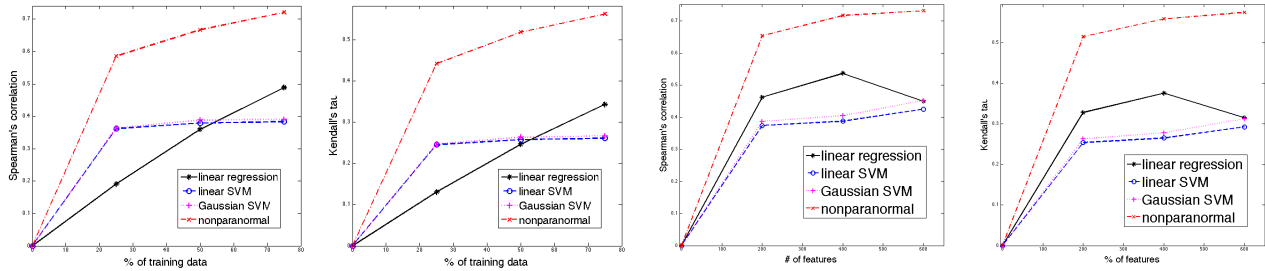


Figure 5: Two figures on the left: varying the amount of training data. L(1): Spearman. L(2): Kendall. Two figures on the right: varying the amount of features. R(1): Spearman. R(2): Kendall.

Datasets	No Dropout	With Dropout
Meme	0.625	<b>0.754*</b>
Finance (pre2009)	0.416	<b>0.482*</b>
Finance (2009)	0.412	<b>0.445*</b>
Finance (post2009)	0.377	<b>0.409*</b>
Meme	0.491	<b>0.580*</b>
Finance (pre2009)	0.307	<b>0.349*</b>
Finance (2009)	0.302	<b>0.318*</b>
Finance (post2009)	0.282	<b>0.297*</b>

Table 2: The effects of dropout training for NPNs on meme and other datasets. The best results of each row are highlighted in **bold**. \* indicates  $p < .001$  comparing to the no dropout setting.

captures the dependency relation of the popular meme series “You mean to tell me...”. Interestingly, the transitional dependency feature  $dep(when, but)$  plays an important role in the language of memes. The object of a preposition, such as  $pobj(vegas, in)$  and  $pobj(life, of)$ , also made the list.

Bigrams are shown to be important features as usual. For example, “Yo daw” is a popular meme based on rapper Xzibit’s famous reality car show “Pimp My Ride”, where the rapper customizes people’s car according to personal preferences. This viral meme follows the pattern<sup>8</sup> of “Yo daw(g), I herd you like  $X$  (noun), so I put an  $X$  in your  $Y$  (noun) so you can  $W$  (verb) while you  $Z$  (verb).”

The use of pronouns, captured by frame semantics features, is associated with popular memes. We hypothesize that by using pronouns such as “i”, “you”, “we”, and “they”, the meme recalls personal experiences and emotions, thus connects better with the audience. Finally, we see that the punctuation bigram “...:” is an important feature in the language

<sup>8</sup><http://knowyourmeme.com/memes/xzibit-yo-dawg>

Top 1-10	Top 11-20	Top 21-30
paul/PER	FE_party_you	new
xcomp(tell,mean)	dep(when,but)	FE_Entity_it
possessive('s,it)	...:	bruce/PER
yo_daw	FE_Theme_i	FE_party_we
pobj(vegas,in)	on_a	FE_Food_fat
ups/ORG	FE_Exp_they	<start>_make
into	FE_Entity_you	so_you
so_you're	<start>_how	penci/PER
FE_Cognizer_i	of_the	y
yo_.	pobj(life,of)	winehouse/PER

Table 3: Top-30 linguistic features that are highly correlated with the popular votes.

of memes, and Web dialect such as “y” (why) also exhibits high correlation with the popular votes.

## 6 Generation Experiments

In this section, we investigate the performance of our meme generation system using 50 test meme images. To quantitatively evaluate our system, we compare with both unsupervised and supervised baselines. For the unsupervised baselines, we compare with a compact recurrent neural network language model (RNNLM) (Mikolov, 2012) trained on the 3,008 text descriptions of our meme training set, as well as a full model of RNNLM trained on a large meme corpus of 269K sentences<sup>9</sup>. For the supervised baselines, all models are trained on the 3,008 training image-description pairs with labels. All these models can be viewed as different re-ranking methods for the retrieved candidate descriptions. We use BLEU score (Papineni et al., 2002) as the evaluation metric, since the generation task can be viewed as translating raw images into sentences, and it is

<sup>9</sup>Note that there are no image features feeding to the unsupervised RNN models.



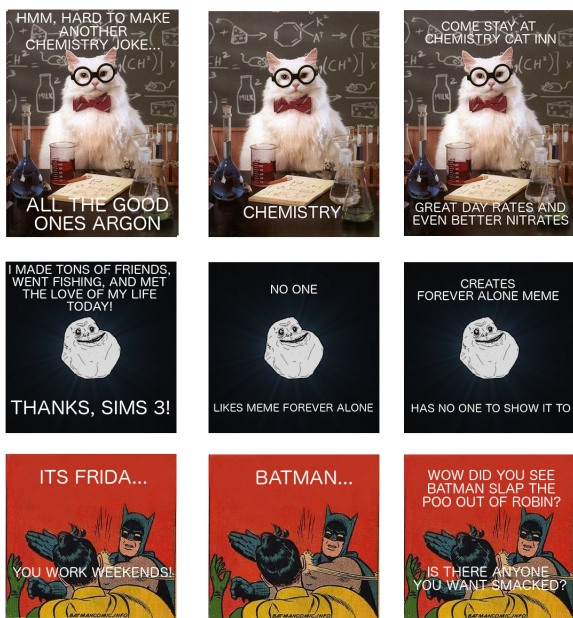


Figure 6: Examples from the meme generation experiment. First row: the chemistry cat meme. Second row: the forever alone meme. Third row: the Batman slaps Robin meme. Left column: human generated top-voted meme descriptions on memegenerator.net at the time of writing. Middle column: generated output from RNNLM. Right column: generated output from NPNs.

used in many caption generation studies (Vinyals et al., 2014; Chen and Zitnick, 2014; Donahue et al., 2014; Fang et al., 2014; Karpathy and Fei-Fei, 2014).

The generation result is shown in Table 4. Note that when combining B-1 to B-4 scores, BLEU includes a brevity penalty as described in the original BLEU paper. We see that our NPN model outperforms the best supervised baseline by 4.35 BLEU points, while also obtaining an advantage of 4.48

Systems	BLEU	B-1	B-2	B-3	B-4
RNN-C	19.52	62.2	21.2	12.1	9.0
RNN-F	23.76	72.2	<b>31.4*</b>	16.2	8.7
LR	23.89	<b>72.3</b>	28.3	15.0	10.6
LSVM	21.06	65.0	24.8	13.1	9.3
GSVM	20.63	66.2	22.8	12.8	9.3
NPN	<b>28.24*</b>	66.9	29.0	<b>19.7*</b>	<b>16.6*</b>

Table 4: The BLEU scores for generating memes from images. B-1 to B-4: BLEU unigram to four-grams. The best BLEU results are highlighted in **bold**. \* indicates  $p < .001$  comparing to the second best system.

BLEU points over the full RNNLM, which is trained on a corpus that is  $\sim 90$  times larger, in an unsupervised fashion. When breaking down the results, we see that our NPN’s advantage is on generating longer phrases, typically trigrams and four-grams, comparing to the other models. This is very interesting, because generating high-quality long phrases is difficult, since the memes are often short.

We show some generation examples in Figure 6. We see that on the left column, the reference memes are the ones with top votes by the crowd. The first *chemistry cat* meme includes puns, the second *forever alone* meme includes reference to the life simulation video game, while the last *Batman* meme has interesting conversations. In the second column, we see that the memes generated by the full RNNLM model are short, which corresponds to the quantitative results in Table 4. In the third column, our NPN meme generator was able to generate longer descriptions. Interestingly, it also creates a pun for the *chemistry cat* meme. Our generation on the *forever alone* meme is also accurate. In the Batman example, we show that the NPN model makes a sentence-image-mismatch type of error: although the generated sentence includes the entities Batman and Robin, as well as their slapping activity, it was originally created for the “overly attached girlfriend” meme<sup>10</sup>.

## 7 Conclusions

In this paper, we study the language of memes by jointly learning the image, the description, and the popular votes. In particular, we propose a robust nonparanormal approach to transform all vision and text features into the cumulative density function space. By learning the stochastic dependencies, we show that our model significantly outperforms various competitive baselines in the prediction experiments. In addition, we also propose a simple pipeline for generating memes from raw images, drawing the wisdom from reverse image search and traditional information retrieval perspectives. Finally, we show that our model obtains significant BLEU point improvements over an unsupervised RNNLM baseline trained on a larger corpus, as well as other strong supervised baselines.

<sup>10</sup><http://www.overlyattachedgirlfriend.com>

## References

- Joshua Albrecht and Rebecca Hwa. 2007. Regression for sentence-level mt evaluation with pseudo references. In *Proceedings of ACL*.
- Yoav Artzi, Patrick Pantel, and Michael Gamon. 2012. Predicting responses to microblog posts. In *Proceedings of NAACL-HLT*.
- Eytan Bakshy, Jake M Hofman, Winter A Mason, and Duncan J Watts. 2011. Everyone’s an influencer: quantifying influence on twitter. In *Proceedings of WSDM*, pages 65–74. ACM.
- Anna Bosch, Andrew Zisserman, and Xavier Munoz. 2007. Image classification using random forests and ferns.
- Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM TIST*.
- Xiaohong Chen and Yanqin Fan. 2006. Estimation of copula-based semiparametric time series models. *Journal of Econometrics*.
- Xinlei Chen and C Lawrence Zitnick. 2014. Learning a recurrent visual representation for image caption generation. *arXiv preprint arXiv:1411.5654*.
- David Christensen. 2005. Fast algorithms for the calculation of kendalls  $\tau$ . *Computational Statistics*.
- Dipanjan Das, Nathan Schneider, Desai Chen, and Noah A Smith. 2010. Probabilistic frame-semantic parsing. In *Proceedings of NAACL-HLT*.
- Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *Proceedings of ACL*.
- Francis X Diebold, Todd A Gunther, and Anthony S Tay. 1997. Evaluating density forecasts.
- Jesse Dodge, Amit Goyal, Xufeng Han, Alyssa Mensch, Margaret Mitchell, Karl Stratos, Kota Yamaguchi, Yejin Choi, Hal Daumé III, Alexander C Berg, et al. 2012. Detecting visual text. In *Proceedings of the NAACL-HLT*.
- Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2014. Long-term recurrent convolutional networks for visual recognition and description. *arXiv preprint arXiv:1411.4389*.
- Carsten Eickhoff, Arjen P. de Vries, and Kevyn Collins-Thompson. 2013. Copulas for information retrieval. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*.
- Charles Elkan. 2003. Using the triangle inequality to accelerate k-means. In *ICML*, volume 3, pages 147–153.
- Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John Platt, et al. 2014. From captions to visual concepts and back. *arXiv preprint arXiv:1411.4952*.
- 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 *Computer Vision—ECCV 2010*, pages 15–29. Springer.
- Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim. 2014. Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, 15:3133–3181.
- Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.
- Alan Gelfand and Adrian Smith. 1990. Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*.
- Zoubin Ghahramani, Barnabás Póczos, and Jeff Schneider. 2012. Copula-based kernel dependency measures. In *Proceedings of the 29th International Conference on Machine Learning*.
- Abhinav Gupta, Praveen Srinivasan, Jianbo Shi, and Larry S Davis. 2009. Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2012–2019. IEEE.
- Fang Han, Tuo Zhao, and Han Liu. 2012. Coda: High dimensional copula discriminant analysis. *Journal of Machine Learning Research*.
- Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.
- Micah Hodosh, Peter Young, and Julia Hockenmaier. 2013. Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Intell. Res.(JAIR)*, 47:853–899.
- Robert V Hogg and Allen Craig. 1994. Introduction to mathematical statistics.
- Liangjie Hong, Ovidiu Dan, and Brian D Davison. 2011. Predicting popular messages in twitter. In *Proceedings of WWW*.
- Harry Joe. 1997. *Multivariate models and dependence concepts*.

- Andrej Karpathy and Li Fei-Fei. 2014. Deep visual-semantic alignments for generating image descriptions. *Stanford University Technical Report*.
- Maurice Kendall. 1938. A new measure of rank correlation. *Biometrika*.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.
- Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. 2011. Baby talk: Understanding and generating image descriptions. In *Proceedings of the 24th CVPR*. Citeseer.
- Mirella Lapata. 2006. Automatic evaluation of information ordering: Kendall’s tau. *Computational Linguistics*.
- Han Liu, John Lafferty, and Larry Wasserman. 2009. The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *The Journal of Machine Learning Research*, 10:2295–2328.
- Han Liu, Fang Han, Ming Yuan, John Lafferty, and Larry Wasserman. 2012. High-dimensional semiparametric gaussian copula graphical models. *The Annals of Statistics*.
- Tomáš Mikolov. 2012. *Statistical language models based on neural networks*. Ph.D. thesis, Ph. D. thesis, Brno University of Technology.
- Margaret Mitchell, Xufeng Han, Jesse Dodge, Alyssa Mensch, Amit Goyal, Alex Berg, Kota Yamaguchi, Tamara Berg, Karl Stratos, and Hal Daumé III. 2012. Midge: Generating image descriptions from computer vision detections. In *Proceedings of EACL*.
- Roger B Nelsen. 1999. *An introduction to copulas*. Springer Verlag.
- Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135.
- Maxime Oquab, Leon Bottou, Ivan Laptev, Josef Sivic, et al. 2013. Learning and transferring mid-level image representations using convolutional neural networks.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318. Association for Computational Linguistics.
- Rahul A Parsa and Stuart A Klugman. 2011. Copula regression. *Variance Advancing and Science of Risk*.
- Emanuel Parzen. 1962. On estimation of a probability density function and mode. *The annals of mathematical statistics*.
- Michael Pitt, David Chan, and Robert Kohn. 2006. Efficient bayesian inference for gaussian copula regression models. *Biometrika*.
- Berthold Schweizer and Abe Sklar. 1983. *Probabilistic metric spaces*.
- Josef Sivic and Andrew Zisserman. 2003. Video google: A text retrieval approach to object matching in videos. In *Proceedings of ICCV*, pages 1470–1477. IEEE.
- Abe Sklar. 1959. *Fonctions de répartition à n dimensions et leurs marges*. Université Paris 8.
- Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*, pages 1631–1642. Citeseer.
- Chenhao Tan, Lillian Lee, and Bo Pang. 2014. The effect of wording on message propagation: Topic- and author-controlled natural experiments on twitter. In *Proceedings of ACL*.
- Robert Tibshirani. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.
- Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of NAACL-HLT*.
- Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2014. Show and tell: A neural image caption generator. *arXiv preprint arXiv:1411.4555*.
- Stefan Wager, Sida Wang, and Percy Liang. 2013. Dropout training as adaptive regularization. In *Advances in Neural Information Processing Systems*, pages 351–359.
- William Yang Wang and Zhenhao Hua. 2014. A semiparametric gaussian copula regression model for predicting financial risks from earnings calls. In *Proceedings of ACL*.
- Sida Wang and Christopher Manning. 2013. Fast dropout training. In *Proceedings of ICML*.
- Dani Yogatama, Michael Heilman, Brendan O’Connor, Chris Dyer, Bryan R Routledge, and Noah A Smith. 2011. Predicting a scientific community’s response to an article. In *Proceedings of EMNLP*.