Understanding Satirical Articles Using Common-Sense

Dan Goldwasser
Purdue University
Department of Computer Science
dgoldwas@purdue.edu

Xiao Zhang
Purdue University
Department of Computer Science
zhang923@purdue.edu

Abstract

Automatic satire detection is a subtle text classification task, for machines and at times, even for humans. In this paper we argue that satire detection should be approached using common-sense inferences, rather than traditional text classification methods. We present a highly structured latent variable model capturing the required inferences. The model abstracts over the specific entities appearing in the articles, grouping them into generalized categories, thus allowing the model to adapt to previously unseen situations.

1 Introduction

Satire is a writing technique for passing criticism using humor, irony or exaggeration. It is often used in contemporary politics to ridicule individual politicians, political parties or society as a whole. We restrict ourselves in this paper to such political satire articles, broadly defined as articles whose purpose is not to report real events, but rather to mock their subject matter. Satirical writing often builds on real facts and expectations, pushed to absurdity to express humorous insights about the situation. As a result, the difference between real and satirical articles can be subtle and often confusing to readers. With the recent rise of social media outlets, satirical articles have become increasingly popular and have famously fooled several leading news agencies[1]. These misinterpretations can often be attributed to careless reading, as there is a clear line between unusual events finding their way to the news and satire, which intentionally places key political figures in unlikely humorous scenarios. The two can be separated by carefully reading the articles, exposing the satirical nature of the events described in such articles.

> Vice President Joe Biden suddenly barged in, asking if anyone could "hook [him] up with a Dixie cup" of their urine. "C'mon, you gotta help me get some clean whiz. Shinseki, Donovan, I'm looking in your direction" said Biden.

> "Do you want to hit this?" a man asked President Barack Obama in a bar in Denver Tuesday night. The president laughed but didn't indulge. It wasn't the only time Obama was offered weed on his night out.

Figure 1: **Examples of real and satirical articles**. **Top:** satirical news excerpt. **Bottom:** real news excerpt.

In this paper we follow this intuition. We look into the satire detection task (Burfoot and Baldwin, 2009), predicting if a given news article is real or satirical, and suggest that this prediction task should be defined over common-sense inferences, rather than looking at it as a lexical text classification task (Pang and Lee, 2008; Burfoot and Baldwin, 2009), which bases the decision on word-level features.

To further motivate this observation, consider the two excerpts in Figure 1. Both excerpts mention top-ranking politicians (the President and Vice President) in a drug-related context, and contain informal slang utterances, inappropriate for the subjects'

[1] https://newrepublic.com/article/118013/satire-news-websites-are-cashing-gullible-outraged-readers

Transactions of the Association for Computational Linguistics, vol. 4, pp. 537–549, 2016. Action Editor: Timothy Baldwin.
Submission batch: 1/2016; Revision batch: 5/2016; Published 12/2016.

position. The difference between the two examples is apparent when analyzing the situation described in the two articles: The first example (top), describes the Vice President speaking inappropriately in a work setting, clearly an unrealistic situation. In the second (bottom) the President is *spoken to* inappropriately, an unlikely, yet not unrealistic, situation. From the perspective of our prediction task, it is advisable to base the prediction on a structured representation capturing the events and their participants, described in the text.

The absurdity of the situation described in satirical articles is often not unique to the *specific* individuals appearing in the narrative. In our example, both politicians are interchangeable: placing the president in the situation described in the first excerpt would not make it less absurd. It is therefore desirable to make a common-sense inference about high-ranking politicians in this scenario.

We follow these intuitions and suggest a novel approach for the satire prediction task. Our model, COMSENSE, makes predictions by making common-sense inferences over a simplified narrative representation. Similarly to prior work (Chambers and Jurafsky, 2008; Goyal et al., 2010; Wang and McAllester, 2015) we represent the narrative structure by capturing the main entities (and tracking their mentions throughout the text), their activities, and their utterances. The result of this process is a Narrative Representation Graph (NRG). Figure 2 depicts examples of this representation for the excerpts in Figure 1.

Given an NRG, our model makes inferences quantifying how likely are each of the represented events and interactions to appear in a real, or satirical context. Annotating the NRG for such inferences is a challenging task, as the space of possible situations is extremely large. Instead, we frame the required inferences as a highly-structured latent variable model, trained discriminatively as part of the prediction task. Without explicit supervision, the model assigns categories to the NRG vertices (for example, by grouping politicians into a single category, or by grouping inappropriate slang utterances, regardless of specific word choice). These category assignments form the infrastructure for higher-level reasoning, as they allows the model to identify the commonalities between unrelated people, their actions and their words. The model learns common-sense patterns leading to real or satirical decisions based on these categories. We express these patterns as parametrized rules (acting as global features in the prediction model), and base the prediction on their activation values. In our example, these rules can capture the combination of ($\mathrm{E}_{Politician}$) $\wedge$ (Q_{slang})$\rightarrow$ Satire, where $\mathrm{E}_{Politician}$ and Q_{slang} are latent variable assignments to entity and utterance categories respectively.

Our experiments look into two variants of satire prediction: using full articles, and the more challenging sub-task of predicting if a quote is real given its speaker. We use two datasets collected 6 years apart. The first collected in 2009 (Burfoot and Baldwin, 2009) and an additional dataset collected recently. Since satirical articles tend to focus on current events, the two datasets describe different people and world events. To demonstrate the robustness of our COMSENSE approach we use the first dataset for training, and the second as out-of-domain test data. We compare COMSENSE to several competing systems including a state-of-the-art Convolutional Neural Network (Kim, 2014). Our experiments show that COMSENSE outperforms all other models. Most interestingly, it does so with a larger margin when tested over the out-of-domain dataset, demonstrating that it is more resistant to overfitting compared to other models.

2 Related Work

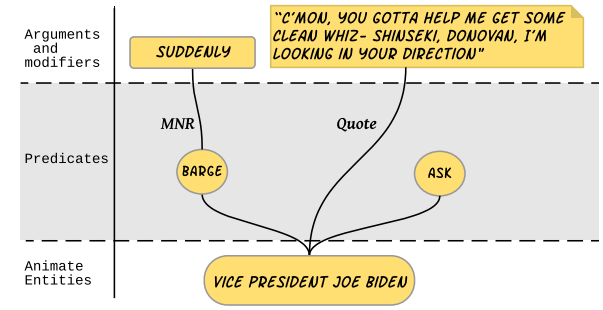
The problem of building computational models dealing with humor, satire, irony and sarcasm has attracted considerable interest in the the Natural Language Processing (NLP) and Machine Learning (ML) communities in recent years (Wallace et al., 2014; Riloff et al., 2013; Wallace et al., 2015; Davidov et al., 2010; Karoui et al., 2015; Burfoot and Baldwin, 2009; Tepperman et al., 2006; González-Ibánez et al., 2011; Lukin and Walker, 2013; Filatova, 2012; Reyes et al., 2013). Most work has looked into ironic expressions in shorter texts, such as tweets and forum comments. Most related to our work is Burfoot and Baldwin (2009) which focused on satirical articles. In that work the authors suggest a text classification approach for satire detection. In addition to using bag-of-words features, the

authors also experiment with semantic validity features which pair entities mentioned in the article, thus capturing combinations unlikely to appear in a real context. This paper follows a similar intuition; however, it looks into structured representations of this information, and studies their advantages.

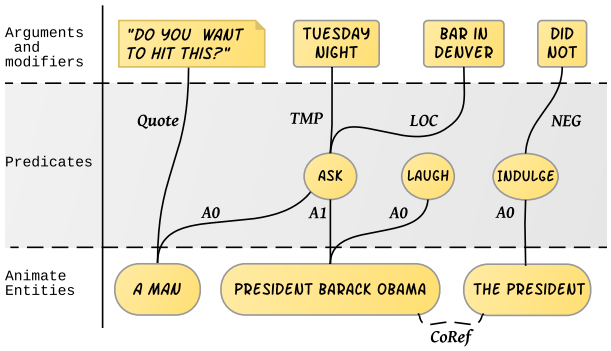
Our structured representation is related to several recent reading comprehension tasks (Richardson et al., 2013; Berant et al., 2014) and work on narrative representation such, as event-chains (Chambers and Jurafsky, 2009; Chambers and Jurafsky, 2008), plot-units (Goyal et al., 2010; Lehnert, 1981) and Story Intention Graphs (Elson, 2012). Unlike these works, narrative representation is not the focus of this work, but rather provides the basis for making inferences, and as result we choose a simpler (and more robust) representation, most closely resembling event chains (Chambers and Jurafsky, 2008)

Making common-sense inferences is one of the core missions of AI, applicable to a wide range of tasks. Early work (Reiter, 1980; McCarthy, 1980; Hobbs et al., 1988) focused on logical inference, and manual construction of such knowledge repositories (Lenat, 1995; Liu and Singh, 2004). More recently, several researchers have looked into automatic common-sense knowledge construction and expansion using common-sense inferences (Tandon et al., 2011; Bordes et al., 2011; Socher et al., 2013; Angeli and Manning, 2014). Several works have looked into combining NLP with common-sense (Gerber et al., 2010; Gordon et al., 2011; LoBue and Yates, 2011; Labutov and Lipson, 2012; Gordon et al., 2012). Most relevant to our work is a SemEval-2012 task (Gordon et al., 2012), looking into common-sense causality identification prediction.

In this work we focus on a different task, satire detection in news articles. We argue that this task is inherently a common-sense reasoning task, as identifying the satirical aspects in narrative text does not require any specialized training, but instead relies heavily on common expectations of normative behavior and deviation from it in satirical text. We design our model to capture these behavioral expectations using (weighted) rules, instead of relying on lexical features as is often the case in text categorization tasks. Other common-sense frameworks typically build on existing knowledge bases representing world knowledge; however, specifying in advance the behaviors commonly associated with people based on their background and situational context, to the extent it can provide good coverage for our task, requires considerable effort. Instead, we suggest to learn this information from data directly, and our model learns jointly to predict and represent the satirical elements of the article.



(a) NRG for a satirical article



(b) NRG for a real article

Figure 2: **Narrative Representation Graph (NRG) for two article snippets**

3 Modeling

Given a news article, our COMSENSE system first constructs a graph-based representation of the narrative, denoted Narrative Representation Graph (NRG), capturing its participants, their actions and utterances. We describe this process in more detail in Section 3.1. Based on the NRG, our model makes a set of inferences, mapping the NRG vertices to general categories abstracting over the specific NRG. These abstractions are formulated as latent variables in our model. The system makes a prediction by reasoning over the abstract NRG, by

decomposing it into paths, where each path captures a partial view of the abstract NRG. Finally we associate the paths with the satire decision output. The COMSENSE model then solves a global inference problem, formulated as an Integer Linear Program (ILP) instance, looking for the most likely explanation of the satire prediction output, consistent with the extracted patterns. We explain this process in detail in Section 3.2.

NRG Abstraction as Common-Sense The main goal of the COMSENSE approach is to move away from purely lexical models, and instead base its decisions on common-sense inferences. We formulate these inferences as parameterized rules, mapping elements of the narrative, represented using the NRG, to a classification decision. The rules' ability to capture common-sense inferences hinges on two key elements. *First*, the abstraction of NRG nodes into typed narrative elements allows the model to find commonalities across entities and their actions. This is done by associating each NRG node with a set of latent variables. *Second*, constructing the decision rules according to the structure of the NRG graph allows us to model the dependencies between narrative elements. This is done by following the paths in the abstract NRG, generating rules by combining the latent variables representing nodes on the path, and associating them with a satire decision variable.

Computational Considerations When setting up the learning system, there is a clear expressivity/efficiency tradeoff over these two elements. Increasing the number of latent variables associated with each NRG node would allow the model to learn a more nuanced representation. Similarly, generating rules by following longer NRG paths would allow the model to condition its satire decision on multiple entities and events jointly. The added expressivity does not come without price. Given the limited supervision afforded to the model when learning these rules, additional expressivity would result in a more difficult learning problem which could lead to overfitting. Our experiments demonstrate this tradeoff, and in Figure 4 we show the effect of increasing the number of latent variables on performance. An additional concern with increasing the model's expressivity is computational efficiency. Satire prediction is formulated as an ILP inference process jointly assigning values to the latent variables and making the satire decision. Since ILP is exponential in the number of variables, increasing the number of latent variables would be computationally challenging. In this paper we take a straight-forward approach to ensuring computational tractability by limiting the length of NRG paths considered by our model to a constant size c=2. Assuming that we have m latent categories associated with each node, each path would generate m^c ILP variables (see Section 3.3 for details), hence the importance of limiting the length of the path. In the future we intend to study approximate inference methods that can help alleviate this computational difficultly, such as using LP-approximation (Martins et al., 2009).

3.1 Narrative Representation Graph for News Articles

The Narrative Representation Graph (NRG) is a simple graph-based representation for narrative text, describing the connections between entities and their actions. The key motivation behind NRG was to provide the structure necessary for making inferences, and as a result we chose a simple representation that does not take into account cross-event relationships, and nuanced differences between some of the event argument types. While other representations (Mani, 2012; Goyal et al., 2010; Elson, 2012) capture more information, they are harder to construct and more prone to error. We will look into adapting these models for our purpose in future work.

Since satirical articles tend to focus on political figures, we design the NRG around animate entities that drive the events described in the text, their actions (represented as predicate nodes), their contextualizing information (location-modifiers, temporal modifiers, negations), and their utterances. We omitted from the graph other non-animate entity types. In Figure 2 we show an example of this representation.

Similar in spirit to previous work (Goyal et al., 2010; Chambers and Jurafsky, 2008), we represent the relations between the entities that appear in the story using a Semantic Role Labeling system (Punyakanok et al., 2008) and collapse all the entity mentions into a single entity using a Co-Reference resolution system (Manning et al., 2014). We attribute

utterances to their speaker based on a previously published rule based system (O'Keefe et al., 2012).

Formally, we construct a graph $G = \{V, E\}$, where V consists of three types of vertices: ANIMATE ENTITY (e.g., people), PREDICATE (e.g., actions) and ARGUMENT (e.g., utterances, locations). The edges E capture the relationships between vertices. The graph contains several different edges. COREF edges collapse the mentions of the same entity into a single entity, ARGUMENT-TYPE edges connect ANIMATE ENTITY nodes to PREDICATE nodes[2], and PREDICATE nodes to argument nodes (modifiers). Finally we add QUOTE edges connecting ANIMATE ENTITY nodes to utterances (ARGUMENT).

3.2 Satire Prediction using the Narrative Representation Graph

Satire prediction is inherently a text classification problem. Such problems are often approached using a Bag-of-Words (BoW) model which ignores the document structure when making predictions. Instead, the NRG provides a structured representation for making the satire prediction. We begin by showing how the NRG can be used directly and then discuss how to enhance it by mapping the graph into abstract categories.

Directly Using NRG for Satire Prediction We suggest a simple approach for extracting features directly from the NRG, by decomposing it into graph paths, *without* mapping the graph into abstract categories. This simple, word-based representation for prediction structured according to the NRG (denoted NARRLEX), generates features by using the words in the original document, corresponding to the graph decomposition. For example, consider the path connecting "a man" to an utterance in Figure 2(b). Simple features could associate the utterances words with that entity, rather than with the President. The resulting NARRLEX model generates Bag-of-Words features based on words corresponding to NRG path vertices, conditioned on their connected entity vertex.

Using Common-Sense for Satire Prediction Unlike the NARRLEX model, which relies on directly observed information, our COMSENSE model performs inference over higher level patterns. In this model the prediction is a global inference process, taking into account the relationships between NRG elements (and their abstraction into categories) and the final prediction. This process is described in Figure 3.

First, the model associates a high level category, that can be reused even when other, previously unseen, entities are discussed in the text. We associate a set of Boolean variables with each NRG vertex, capturing higher level abstraction over this node.

We define three types of categories corresponding to the three types of vertices, and denote them E, A, Q for Entity category, Action category and Quote category, respectively. Each category variable can take k different values. As a convention we denote $X = i$ as category assignment, where $X \in \{E, A, Q\}$ is the category type, and i is its assignment. Since these category assignments are not directly observed, they are treated as latent variables in our model. This process is exemplified at the top right corner of Figure 3.

Combinations of category assignments form patterns used for determining the prediction. These patterns can be viewed as parameterized rules. Each weighted rule associates a combination with an output variable (SATIRE or REAL). Examples of such rules are provided in the middle of the right corner of Figure 3. We formulate the activations of these rules as Boolean variables, whose assignments are highly interconnected. For example, the variables representing the following rules $(E = 0) \rightarrow$ SATIRE and $(E = 0) \rightarrow$ REAL are mutually exclusive, since assigning a T value to either one entails a satire (or real) prediction. To account for this interdependency, we add constraints capturing the relations between rules.

The model makes predictions by combining the rule weights and predicting the top scoring output value. The prediction can be viewed as a derivation process, mapping article entities to categories (e.g., ENTITY("A MAN")$\rightarrow$ (E=0), is an example of such derivation), combinations of categories compose into prediction patterns (e.g., (E=0)$\rightarrow$SATIRE). We use an ILP solver to find the optimal derivation sequence. We describe the inference process as an Integer Linear Program in the following section.

[2]These edges are typed according to their semantic roles.

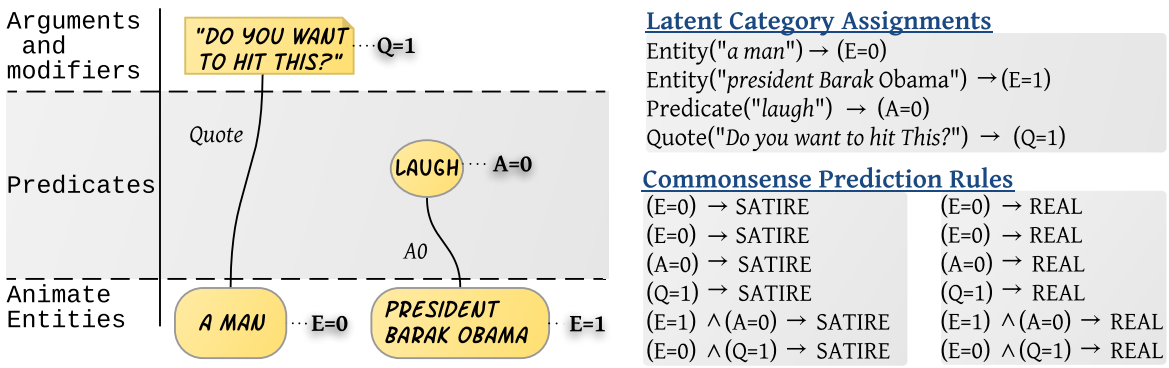


Figure 3: **Extracting Common-sense prediction rules.**

3.3 Identifying Relevant Interactions using Constrained Optimization

We formulate the decision as a 0-1 Integer Linear Programming problem, consisting of three types of Boolean variables: category assignments indicator variables, indicator variables for common-sense patterns, and finally the output decision variables. Each indicator variable is also represented using a feature set, used to score its activation.

3.3.1 Category Assignment Variables

Each node in the NRG is assigned a set of competing variables, mapping the node to different categories according to its type.

- **ANIMATE ENTITY Category Variables**, denoted $h_{i,j,E}$, indicating the Entity category *i* for NRG vertex *j*.
- **ACTION Category Variables**, denoted $h_{i,j,A}$, indicating the Action category *i* for NRG vertex *j*.
- **QUOTE Category Variables**, denoted $h_{i,j,Q}$, indicating the Quote category *i* for NRG vertex *j*.

  The number of possible categories for each variable type is a hyper-parameter of the model.

Variable activation constraints Category assignments to the same node are mutually exclusive (a node can only have a single category). We encode this fact by constraining the decision with a linear constraint (where X $\in \{E, A, Q\}$):

$$\forall j \sum_{i} \quad h_{i,j,X} = 1.$$

Category Assignment Features Each decision variable decomposes into a set of features, $\phi(\mathbf{x}, h_{i,j,X})$ capturing the words associated with the *j*-th vertex, conditioned on X and *i*.

3.3.2 Common-sense Patterns Variables

We represent common-sense prediction rules using an additional set of Boolean variables, connecting the category assignments variables with the output prediction. The space of possible variables is determined by decomposing the NRG into paths of size up to 2, and associating two Boolean variables with category assignment variables corresponding to the vertices on these paths. One of the variables associates the sequence of category assignment variables with a REAL output value, and one with a SATIRE output value.

- **Single Vertex Path Patterns Variables**, denoted by $h^B_{h_{i,j,X}}$, indicating that the category assignment captured by $h_{i,j,X}$ is associated with output value B (where B$\in \{$SATIRE, REAL$\}$).
- **Two Vertex Path Patterns Variables**, denoted by $h^B_{(h_{i,j,X_1}),(h_{k,l,X_2})}$, indicating that the pattern captured by category assignment along the NRG path of h_{i,j,X_1} and h_{i,j,X_2} is associated with output value B (where B$\in \{$SATIRE, REAL$\}$).

Decision Consistency constraints It is clear that the activation of the common-sense Patterns Variables entails the activation of the category assignment variables, corresponding to the elements of the common-sense patterns. For readability we only write the constraint for the Single Vertex Path Variables:

$$(h^B_{h_{i,j,X}}) \implies (h_{i,j,X}).$$

Features Similar to the category assignment variable features, each decision variable decomposes into a set of features, $\phi(\mathbf{x}, h^B_{h_{i,j,X}})$. These features captures the words associated with each of the category assignment variables (in this example, the words associated with the *j*-th vertex) conditioned on the category assignments and the output prediction value (in this example, *X, i* and *B*). We also add a feature $\phi(h_{i,j,X}, B)$ capturing the connection between the output value B, and category assignment.

3.3.3 Satire Prediction Variables

Finally, we add two more Boolean variables corresponding to the output prediction: h^{Satire} and h^{Real}. The activation of these two variables is mutually exclusive, we encode that by adding the constraint:

$$h^{Satire} + h^{Real} = 1.$$

We ensure the consistency of our model adding constraints forcing agreement between the final prediction variables, and the common-sense patterns variables:

$$h^B_{h_{i,j,X}} \implies h^B.$$

Overall Optimization Function

The Boolean variables described in the previous section define a space of competing inferences. We find the optimal output value derivation by finding the optimal set of variables assignments, by solving the following objective:

$$\max_{y,\mathbf{h}} \sum_i h_i \mathbf{w}^T \phi(\mathbf{x}, h_i, y)$$
$$s.t. \quad \mathbf{C}, \quad \forall i; h_i \in \{0,1\}, \tag{1}$$

where $h_i \in H$ is the set of all variables defined above and $\mathbf{C}$ is the set of constraints defined over the activation of these variables. $\mathbf{w}$ is the weight vector, used to quantify the feature representation of each h, obtained using a feature function $\phi(\cdot)$.

Note that the Boolean variable acts as a 0-1 indicator variable. We formalize Eq. (1) as an ILP instance, which we solve using the highly optimized Gurobi toolkit[3].

[3] http://www.gurobi.com/

4 Parameter Estimation for COMSENSE

The COMSENSE approach models the decision as interactions between high-level categories of entities, actions and utterances. However, the high level categories assigned to the NRG vertices are not observed, and as a result we view it as a weakly supervised learning problem, where the category assignments correspond to latent variable assignments. We learn the parameters of these assignments by using a discriminative latent structure learning framework.

The training data is a collection $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where $\mathbf{x}_i$ is an article, parsed into an NRG representation, and y is a binary label, indicating if the article is satirical or real.

Given this data we estimate the models' parameters by minimizing the following objective function.

$$L_D(\mathbf{w}) = \min_w \frac{\lambda}{2}||\mathbf{w}||^2 + \frac{1}{n}\sum_{i=1}^n \xi_i \tag{2}$$

ξ_i is the slack variable, capturing the margin violation penalty for a given training example, and defined as follows:

$$\xi_i = \max_{y,\mathbf{h}} f(\mathbf{x}, \mathbf{h}, y, \mathbf{w}) + cost(y, y_i)$$
$$- \max_{\mathbf{h}} f(\mathbf{x}, \mathbf{h}, y_i, \mathbf{w}),$$

where $f(\cdot)$ is a scoring function, similar to the one used in Eq. 1. The cost function is the margin that the true prediction must exceed over the competing label, and it is simply defined as the difference between the model prediction and the gold label. This formulation is an extension of the hinge loss for latent structure SVM. λ is the regularization parameter controlling the tradeoff between the l_2 regularizer and the slack penalty.

We optimize this objective using the stochastic sub-gradient descent algorithm (Ratliff et al., 2007; Felzenszwalb et al., 2009). We can compute the sub-gradient as follows:

$$\nabla L_D(\mathbf{w}) = \lambda \mathbf{w} + \sum_{i=1}^n \Phi(\mathbf{x}_i, y_i, y^*)$$
$$\Phi(\mathbf{x}_i, y_i, y^*) = \phi(\mathbf{x}_i, \mathbf{h}^*, y_i) - \phi(\mathbf{x}_i, \mathbf{h}^*, y^*),$$

where $\phi(x_i, \mathbf{h}^*, y^*)$ is the set of features representing the solution obtained after solving Eq. 1[4] and

[4] modified to accommodate the margin constraint

making a prediction. $\phi(x_i, \mathbf{h}^*, y_i)$ is the set of features representing the solution obtained by solving Eq. 1 while fixing the outcome of the inference process to the correct prediction (i.e., y_i). Intuitively, it can be considered as finding the best explanation for the correct label using the latent variables $\mathbf{h}$.

In the stochastic version of the sub gradient descent algorithm we approximate $\nabla L_D(\mathbf{w})$ by computing the sub gradient of a single example and making a local update. This version resembles the latent-structure perceptron algorithm (Sun et al., 2009). We repeatedly iterate over the training examples and for each example, if the current $\mathbf{w}$ leads to a correct prediction (and satisfies the margin constraint), we only shrink $\mathbf{w}$ according to λ. If the model makes an incorrect prediction, the model is updated according $\Phi(\mathbf{x}_i, y_i, y^*)$. The optimization objective $L_D(W)$ is not convex, and the optimization procedure is guaranteed to converge to a local minimum.

5 Empirical Study

We design our experimental evaluation to help clarify several questions. First, we want to understand how our model compares with traditional text classification models. We hypothesize that these methods are more susceptible to overfitting, and design our experiments accordingly. We compare the models' performance when using in-domain data (test and training data are from the same source), and out-of-domain data, where the test data is collected from a different source. We look into two tasks. One is the Satire detection task (Burfoot and Baldwin, 2009). We also introduce a new task, called "*did I say that?*" which only focuses on utterances and speakers.

The second aspect of our evaluation focuses on the common-sense inferences learned by our model. We examine how the size of the set of categories impacts the model performance. We also provide a qualitative analysis of the learned categories using a heat map, capturing the activation strength of learned inferences over the training data.

Prediction tasks We look into two prediction tasks: (1) Satire Detection (denoted SD), a binary classification task, in which the model has access to the complete article (2) "*Did I say that?*" (denoted DIST), a binary classification task, consisting only of entities mentions (and their surrounding context in text) and direct quotes. The goal of the DIST is to predict if a given utterance is likely to be real, given its speaker. Since not all document contain direct quotes, we only use a subset of the documents in the SD task.

Datasets In both prediction tasks we look into two settings: (1) In-domain prediction: where the training and test data are collected from the same source, and (2) out-of-domain prediction, where the test data is collected from a different source. We use the data collected by Burfoot and Baldwin (2009) for training the model in both settings, and its test data for in-domain prediction (denoted TRAIN - SD'09, TEST - SD'09, TRAIN - SD'09 - DIST, TEST - SD'09 - DIST, respectively for training and testing in the SD and DIST tasks). In addition, we collected a second dataset of satirical and real articles (denoted SD'16). This collection of articles contains real articles from `cnn.com` and satirical articles from `theonion.com`, a well known satirical news website. The articles were published between 2010 to 2015, appearing in the political sections of both news websites. Following other work in the field, all datasets are highly skewed toward the negative class (real articles), as it better characterizes a realistic prediction scenario. The statistics of the datasets are summarized in Table 2.

Evaluated Systems We compare several systems, as follows:

System	
ALLPOS	Always predict Satire
BB'09	Results by (Burfoot and Baldwin, 2009)
CONV	Convolutional NN. We followed (Kim, 2014), using pre-trained 300-dimensional word vectors (Mikolov et al., 2013).
LEX	SVM with unigram (LEX_U) or both unigram and bigram ($\text{LEX}_\text{U+B}$) features
NARRLEX	SVM with direct NRG-based features (see Sec 3.2)
COMSENSE	Our model. We denote the full model as COMSENSE_F, and COMSENSE_Q when using only the entity+quotes based patterns.

We tuned all the models' hyper-parameters by using a small validation set, consisting of 15% of the training data. After setting the hyper-parameters, the

model was retrained using the entire dataset. We used SVM-light[5] to train our lexical baseline systems (LEX and NARRLEX). Since the data is highly skewed towards the negative class (REAL), we adjust the learner objective function cost factor for positive examples to outweigh negative examples. The cost factor was tuned using the validation set.

5.1 Experimental Results

Since our goal is to identify satirical articles, given significantly more real articles, we report the F-measure of the positive class. The results are summarized in Tables 1 and 3. We can see that in all cases the COMSENSE model obtains the best results. We note that in both tasks, when learning in the out-of-domain settings performance drops sharply, however the gap between the COMSENSE model and other models increases in these settings, showing that it is less prone to overfitting.

Interestingly, for the satire detection (SD) task, the COMSENSE$_Q$ model performs best for the in-domain setting, and COMSENSE$_F$ gives the best performance in the out-of-domain settings. We hypothesize that this is due to a phenomenon we call *"overfitting to document structure"*. Lexical models tend to base the decision on word choices specific to the training data, and as a result when tested on out of domain data, which describes new events and entities, performance drops sharply. Instead, the COMSENSE$_Q$ model focuses on properties of quotations and entities appearing in the text. In the SD'09 datasets, this information helps focus the learner, as the real and satire articles are structured differently (for example, satire articles frequently contain multiple quotes). This structure is not maintained when working with out-of-domain data, and indeed in these settings the model benefits from using additional information offered by the full model.

Number of Latent Categories Our COMSENSE model is parametrized with the number of latent categories it considers for each entity, predicate and quote. This hyper-parameter can have a strong influence on the model performance (and running time). Increasing it adds to the model's expressivity allowing it to learn more complex patterns, but also defines a more complex learning problem (recall our non-convex learning objective function). We focused on the DIST task when evaluating different configurations as it converged much faster than the full model. Figure 4 plots the model behavior when using different numbers of latent categories. Interestingly, the number of entity categories saturates faster than the number of quote categories. This can be attributed to the limited text describing entities.

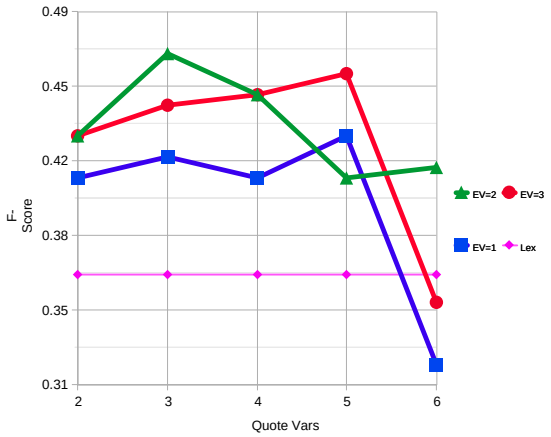


Figure 4: **Different Number of Latent Categories.** EV denotes the number entity categories used, and QuoteVars denotes the number of quote categories used.

Visualizing Latent COMSENSE Patterns Given the assignment to latent categories, our model learns common-sense patterns for identifying satirical and real articles based on these categories. Ideally, these patterns could be extracted directly from the data, however providing the resources for this additional prediction task is not straightforward. Instead, we view the category assignment as latent variables, which raises the question - *what are the categories learned by the model?*

In this section we provide a qualitative evaluation of these categories and the prediction rules identified by the system using the heat map in Figure 5. For simplicity, we focus on the DIST task, which only has categories corresponding to entities and quotes.

(a) Prediction Rules These patterns are expressed as rules, mapping category assignments to output values. In the DIST task, we consider combinations of entity and quote category pairs, denoted Ei,Qj, in the heat map. The top part of Figure 5, in red, shows the activation strength of each of the category com-

[5] http://svmlight.joachims.org/

Task: SD	InDomain (SD'09+SD'09)			OutDomain (SD'09+SD'16)		
	P	R	F	P	R	F
AllPos	0.063	1	0.118	0.121	1	0.214
BB'09	0.945	0.690	0.798	-	-	-
Conv	0.822	0.531	0.614	0.517	0.310	0.452
Lex_{U}	0.920	0.690	0.790	0.298	0.579	0.394
$\text{Lex}_{\text{U+B}}$	0.840	0.720	0.775	0.347	0.367	0.356
NarrLex	0.690	0.590	0.630	0.271	0.425	0.330
ComSense_{Q}	0.839	0.780	**0.808**	0.317	0.706	0.438
ComSense_{F}	0.853	0.70	0.770	0.386	0.693	**0.496**

Table 1: **Results for the SD task**

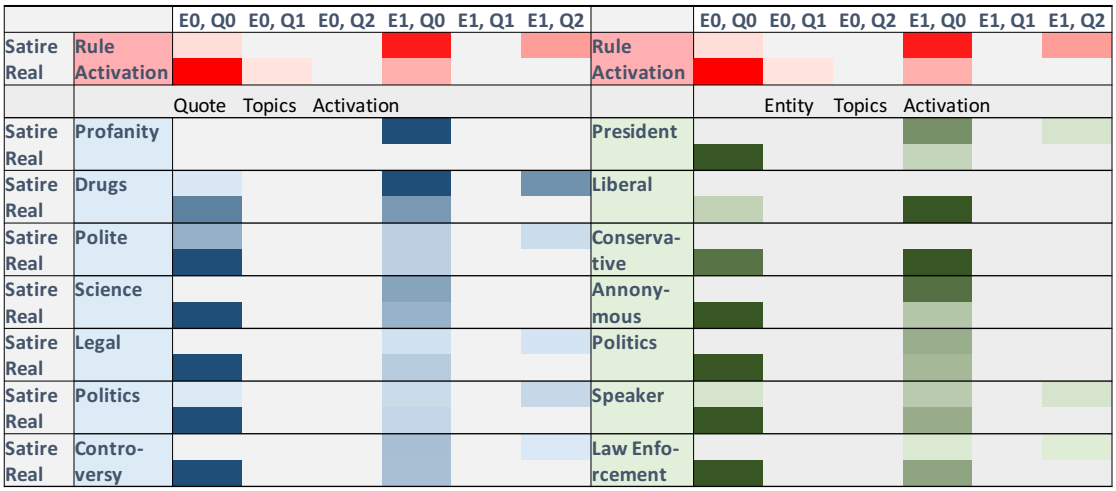


Figure 5: **Visualization of the categories learned by the models.** Color coding capture the activation strength of manually constructed topical word groups, according to each latent category. Darker colors indicate higher values. E_i (Q_i), indicates an entity (Quote) variable assigned the *i*-th category.

Data	Real	Satire
Train - SD'09	2505	133
Test - SD'09	1495	100
Test - SD'16	3117	433
Train - SD'09 - DIST	1160	112
Test - SD'09 - DIST	680	85
Test - SD'16- DIST	1964	362

Table 2: **Datasets statistics**.

binations when making predictions over the training data. Darker colors correspond to larger values, which were computed as:

$$cell(C_E, C_Q, B) = \frac{\sum_j h^B_{(h_{C_E,j,E}),(h_{C_Q,j,Q})}}{\sum_{j,k,l} h^B_{(h_{k,j,E}),(h_{l,j,Q})}}$$

Intuitively, each cell value in Figure 5 is the number of times each category pattern appeared in Real or Satire output predictions, normalized by the overall number of pattern activations for each output.

We assume that different patterns will be associated with satirical and real articles, and indeed we can see that most entities and quotes appearing in Real articles fall into a distinctive category pattern, E_0, Q_0. Interestingly, there is some overlap between the two predictions in the most active Satire category (E_1, Q_0). We hypothesize that this is due to the fact that the two article types have some overlap.

(b) Associating topic words with learned categories In order to understand the entity and quote categories emerging from the training phase, we look at the activation strength of each category pattern with respect to a set of topic words. We manually identified a set of entity types and quote topics, which are likely to appear in political articles. We associate a list of words with each one of these types.

Task: DIST	InDomain (DIST'09+DIST'09)			OutDomain (DIST'09+DIST'16)		
	P	R	F	P	R	F
AllPos	0.110	1	0.198	0.155	1	0.268
Lex_U	0.837	0.423	0.561	0.407	0.328	0.363
ComSense_Q	0.712	0.553	**0.622**	0.404	0.561	**0.469**

Table 3: **Results for the DIST task**

For example, the entity topic PRESIDENT was associated with words such as *president, vice-president, Obama, Biden, Bush, Clinton*. Similarly, we associated with the quote topic PROFANITY a list of profanity words. We associate 7 types with quote categories corresponding to style and topic, namely PROFANITY, DRUGS, POLITENESS, SCIENCE, LEGAL, POLITICS, CONTROVERSY, and another set of seven types with entity types, namely PRESIDENT, LIBERAL, CONSERVATIVE, ANONYMOUS, POLITICS, SPEAKER, LAW ENFORCEMENT.

In the bottom left part of Figure 5 (in blue), we show the activation strength of each category with respect to the set of selected quote topics. Intuitively, we count the number of times the words associated with a given topic appeared in the text span corresponding to a category assignment pair, separately for each output prediction. We normalize this value by the total number of topic word occurrences, over all category assignment pairs. Note that we only look at the text span corresponding to quote vertices in the NRG. We provide a similar analysis for entity categories in the bottom right part of Figure 5 (in green). We show the activation strength of each category with respect to the set of selected entity topic words. As can be expected, we can see that profanity words are only associated with satirical categories, and even more interestingly, when words appear in both satirical and real predictions, they tend to fall into different categories. For example, the topic words related to DRUGS can appear both in real articles discussing alcohol and drug policies. But topic words related to drugs also appear in satirical articles portraying politicians using these substances. While these are only qualitative results, we believe they provide strong intuitions for future work, especially considering the fact that the activation values do not rely on direct supervision, and only reflect the common-sense patterns emerging from the learned model.

6 Summary and Future Work

In this paper we presented a latent variable model for satire detection. We followed the observation that satire detection is inherently a semantic task and modeled the common-sense inferences required for it using a latent variable framework.

We designed our experiments specifically to examine if our model can generalize better than unstructured lexical models by testing it on out-of-domain data. Our experiments show that in these challenging settings, the performance gap between our approach and the unstructured models increases, demonstrating the effectiveness of our approach.

In this paper we restricted ourselves to limited narrative representation. In the future we intend to study how to extend this representation to capture more nuanced information.

Learning common-sense representation for prediction problems has considerable potential for NLP applications. As the NLP community considers increasingly challenging tasks focusing on semantic and pragmatic aspects, the importance of finding such common-sense representation will increase. In this paper we demonstrated the potential of common-sense representations for one application. We hope these results will serve as a starting point for other studies in this direction.

References

Gabor Angeli and Christopher D Manning. 2014. Naturalli: Natural logic inference for common sense reasoning. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Jonathan Berant, Vivek Srikumar, Pei-Chun Chen, Abby Vander Linden, Brittany Harding, Brad Huang, Peter Clark, and Christopher D. Manning. 2014. Modeling biological processes for reading comprehension. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Antoine Bordes, Jason Weston, Ronan Collobert, and Yoshua Bengio. 2011. Learning structured embed-

dings of knowledge bases. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*.

Clint Burfoot and Timothy Baldwin. 2009. Automatic satire detection: Are you having a laugh? In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proc. of the Annual Conference on Computational Natural Language Learning (CoNLL)*.

David K Elson. 2012. Dramabank: Annotating agency in narrative discourse. In *Proc. of the International Conference on Language Resources and Evaluation (LREC)*.

P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. 2009. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(1).

Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Proc. of the International Conference on Language Resources and Evaluation (LREC)*.

Matt Gerber, Andrew S Gordon, and Kenji Sagae. 2010. Open-domain commonsense reasoning using discourse relations from a corpus of weblog stories. In *Proc. of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*.

Roberto González-Ibánez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: a closer look. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*. Association for Computational Linguistics.

Andrew S Gordon, Cosmin Adrian Bejan, and Kenji Sagae. 2011. Commonsense causal reasoning using millions of personal stories. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*.

Andrew S Gordon, Zornitsa Kozareva, and Melissa Roemmele. 2012. Semeval-2012 task 7: choice of plausible alternatives: an evaluation of commonsense causal reasoning. In *Proc. of the Sixth International Workshop on Semantic Evaluation*.

Amit Goyal, Ellen Riloff, and Hal Daumé III. 2010. Automatically producing plot unit representations for narrative text. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Jerry R Hobbs, Mark Stickel, Paul Martin, and Douglas Edwards. 1988. Interpretation as abduction. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Jihen Karoui, Farah Benamara, V?ronique Moriceau, Nathalie Aussenac-Gilles, and Lamia Hadrich Belguith. 2015. Towards a contextual pragmatic model to detect irony in tweets. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Igor Labutov and Hod Lipson. 2012. Humor as circuits in semantic networks. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Wendy G. Lehnert. 1981. Plot units and narrative summarization. *Cognitive Science*, 5(4):293–331.

Douglas B Lenat. 1995. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38.

Hugo Liu and Push Singh. 2004. Conceptnet?a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226.

Peter LoBue and Alexander Yates. 2011. Types of common-sense knowledge needed for recognizing textual entailment. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Stephanie Lukin and Marilyn Walker. 2013. Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. In *Proc. of the Workshop on Language Analysis in Social Media*.

Inderjeet Mani. 2012. *Computational Modeling of Narrative*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

André FT Martins, Noah A Smith, and Eric P Xing. 2009. Polyhedral outer approximations with application to natural language parsing. In *Proc. of the International Conference on Machine Learning (ICML)*.

J. McCarthy. 1980. Circumscription a form of non-monotonic reasoning. *Artificial Intelligence*, 13(1,2).

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *The Conference on Advances in Neural Information Processing Systems (NIPS)*.

Tim O'Keefe, Silvia Pareti, James R Curran, Irena Koprinska, and Matthew Honnibal. 2012. A sequence labelling approach to quote attribution. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

V. Punyakanok, D. Roth, and W. Yih. 2008. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2).

Nathan D. Ratliff, J. Andrew Bagnell, and Martin Zinkevich. 2007. (approximate) subgradient methods for structured prediction. In *Proc. of the International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Raymond Reiter. 1980. A logic for default reasoning. *Artificial intelligence*, 13(1):81–132.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation*, 47(1):239–268.

Matthew Richardson, Christopher J. C. Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *The Conference on Advances in Neural Information Processing Systems (NIPS)*.

Xu Sun, Takuya Matsuzaki, , Daisuke Okanohara, and Junichi Tsujii. 2009. Latent variable perceptron algorithm for structured classication. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Niket Tandon, Gerard De Melo, and Gerhard Weikum. 2011. Deriving a web-scale common sense fact database. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*.

Joseph Tepperman, David R Traum, and Shrikanth Narayanan. 2006. " yeah right": sarcasm recognition for spoken dialogue systems. In *Proc. of Interspeech*.

Byron C. Wallace, Do Kook Choe, Laura Kertz, and Eugene Charniak. 2014. Humans require context to infer ironic intent (so computers probably do, too). In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Byron C. Wallace, Do Kook Choe, and Eugene Charniak. 2015. Sparse, contextually informed models for irony detection: Exploiting user communities, entities and sentiment. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.

Hai Wang and Mohit Bansal Kevin Gimpel David McAllester. 2015. Machine comprehension with syntax, frames, and semantics. In *Proc. of the Annual Meeting of the Association Computational Linguistics (ACL)*.