Effects of Semantic Relatedness between Setups and Punchlines in Twitter Hashtag Games

Andrew Cattle **Xiaojuan Ma**
Hong Kong University of Science and Technology
Department of Computer Science and Engineering
Clear Water Bay, Hong Kong
{acattle,mxj}@cse.ust.hk

Abstract

This paper explores humour recognition for Twitter-based hashtag games. Given their popularity, frequency, and relatively formulaic nature, these games make a good target for computational humour research and can leverage Twitter likes and retweets as humour judgments. In this work, we use pairwise relative humour judgments to examine several measures of semantic relatedness between setups and punchlines on a hashtag game corpus we collected and annotated. Results show that *perplexity*, *Normalized Google Distance*, and *free-word association-based features* are all useful in identifying "funnier" hashtag game responses. In fact, we provide empirical evidence that funnier punchlines tend to be more obscure, although more obscure punchlines are not necessarily rated funnier. Furthermore, the asymmetric nature of free-word association features allows us to see that while punchlines should be harder to predict given a setup, they should also be relatively easy to understand in context.

1 Introduction

Humour is ubiquitous in everyday language and important in social interactions. This has been recognized by the computing industry, as Google recently hired professional jokes writers to help make an upcoming AI assistant seem more natural (Stein, 2016). Beyond their applications in user interfaces (Morkes et al., 1998), the automatic identification, processing, or generation of humour also has applications in diverse fields such as sentiment analysis (Davidov et al., 2010) and computer-aided language acquisition (Ritchie et al., 2007).

While research into computational humour, and humour recognition in particular, has focused on humour as a classification task, humour recognition as a ranking task has received increased attention as of late. To this end, and to develop a more complete model of computational humour, this paper seeks to gain insights into the role of semantic relatedness between punchline and setup and its effects on perceived funniness. Specifically, we examine the semantic relationships between hashtag prompts (setups) and punchlines in Twitter hashtag games. We begin by introducing the task of humour recognition for Twitter hashtag games and describing the creation of an annotated hashtag game corpus. We then introduce multiple semantic relatedness measures including, to the best of our knowledge, the first uses of free word association datasets and Normalized Google Distance in computational humour. We evaluate the predictive power of these semantic relatedness measure for identifying the funnier or a pair of tweets. And finally we derive insights from our results. Although we will limit our analysis to a specific type of hashtag game, semantic relatedness should play a role in almost all humour.

Intuitively, punchlines should be relevant to setups, otherwise they become random non-sequiturs and thus are not funny. Therefore, we expect that punchlines which are very weakly semantically related to their setups will be judged as less humorous since the relevance of the punchline to the setup may be less readily apparent. Conversely, punchlines intuitively should not be obvious. Thus we expect that punchlines which are very strongly semantically related to their setups will be judged as less humorous since the punchline may be too straightforward.

Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media, pages 70–79, Osaka, Japan, December 12 2016.

Ice, Ice Hockey #OlympicSongs @midnight
Smells Like Teen Sprint #OlympicSongs
Should I Sail? or Should I row? #OlympicSongs
I want to know what luge is #OlympicSongs @midnight
I'll tell you what I want, what I relay, relay want #OlympicSongs @midnight

Table 1: Sample responses for #OlympicSongs

Hashtag games, also known as hashtag wars, are a collaborative form of online play which is popular on social media sites, most notably Twitter. They work as follows. Participants write short humorous texts based around a common theme or topic, denoted using a hashtag. By including the common hashtag in their responses, participants can easily see each others responses in almost real time. Participants then compete to see who can come up with the funniest responses and amass the most likes and retweets (Sheridan, 2011). A sampling of responses to the hashtag #OlympicSongs is shown in Table 1. Although the games themselves date back to at least 2011, they have been popularized in recent years by the Comedy Central show *@midnight* through their nightly "Hashtag Wars" segment. These games present an attractive target for computational humour research because of their short length, high popularity, and relatively formulaic nature.

The types of hashtag prompts used in hashtag games are quite diverse. For example, #CollegeIn5Words and #MyLoveLifeIn3Words ask participants to describe a topic in a humorous way using a specified word limit. Other hashtags, such as #WhenIWasYourAge or #WrongReasonsToHaveKids, are even more open-ended as they specify a topic but do not place any other restrictions on responses.

This paper focuses on one of the most common genres of hashtag game in which participants take words or phrases associated with a source domain and modify them to include references to a target domain. For example, #OlympicSongs encourages participants to take song titles, song lyrics, etc. (the source domain) and modify them to include references to the Olympic Games (the target domain), as shown in Table 1. The formulaic nature of such hashtags makes them well suited for computational humour-related research, especially for investigating the relationships between punchlines and their setups. Typically, such modifications result in a pun, such as substituting "relay" for the phonetically similar "really" in the lyrics of the Spice Girls song "Wannabe" or substituting "sprint" for the orthographically similar "spirit" in the title of the Nirvana song "Smells like Teen Spirit". While the quality of such word play undoubtedly affects the perceived funniness of a tweet, this is beyond the scope of this paper.

2 Previous Work

Early work on computational humour focused more on humour generation in specific contexts, such as punning riddles (Binsted and Ritchie, 1994; Ritchie et al., 2007), humorous acronyms (Stock and Strapparava, 2002), or jokes in the form of "I like my X like I like my Y" (Petrovic and Matthews, 2013). Labutov and Lipson (2012) offered a slightly more generalized approach using Semantic Script Theory of Humour.

Recently, humour recognition has gained increasing attention. Taylor and Mazlack (2004) presented a method for recognizing wordplay in "Knock Knock" jokes. Mihalcea and Strapparava (2005) identified stylistic features, such as alliteration and antonymy, to identify humorous one-liners. Mihalcea and Pulman (2007) expanded on this approach, finding that human-centeredness and negative sentiment are both useful in identifying humorous one-liners as well as distinguishing satirical news articles from genuine ones. There is also the related task of irony identification (Davidov et al., 2010; Tsur et al., 2010; Reyes et al., 2012), which typically uses n-gram and sentiment features to distinguish ironic tweets from non-ironic ones.

Although humour recognition has by and large been presented as a classification task, Shahaf et

al. (2015) and Radev et al. (2016) instead reframe humour recognition as a ranking task. Both works aim to identify the funnier of a pair of cartoon captions taken from submissions to The New Yorker's weekly Cartoon Caption Contest[1]. Each week, New Yorker readers are presented "a cartoon in need of a caption" and encouraged to submit their own humorous suggestions. Shahaf et al. (2015) found that simpler grammatical structures, less reliance on proper nouns, and shorter joke phrases all lead to funnier captions. Radev et al. (2016) showed that in addition to human-centeredness and sentiment, high LexRank score was a strong indication of humour, where LexRank is a graph-based text summarization technique introduced in Erkan and Radev (2004).

Works on cartoon caption contests serve as a logical starting point for hashtag game-related research. In both cases participants, who are members of the general public, are supplied with a common prompt which all submissions must relate to. In both cases submissions are short, humorous texts. As such, computational humour techniques designed for cartoon caption contests should be almost directly applicable to hashtag games.

Cartoon caption contests and hashtag games are similar in other ways, too. Both gather a large number of submissions; an average of 4,808 captions per cartoon (Shahaf et al., 2015) versus 11,278 responses per hashtag. Shahaf et al. (2015) and Radev et al. (2016) both noted that cartoon captions tended to hinge on similar jokes. While hashtag game responses also tended to hinge on similar jokes, this appeared to occur at a lower rate than in cartoon caption data, potentially due to hashtag game responses being visible to all participants as opposed to cartoon caption contests' closed submission system.

Despite their similarities, hashtag games offer several advantages over cartoon caption contests. First, setups are denoted using text-based hashtags, meaning they can be processed in a similar way to the responses. By comparison, cartoons, being a visual medium, require computer vision techniques in order to automatically extract setup-related features, adding system complexity. Furthermore, computer vision techniques are not yet sophisticated enough to reliably extract such features. This is why Shahaf et al. (2015) resorted to human annotations in order to extract context information from the cartoon prompts. Second, while works on cartoon captions have relied on Amazon Mechanical Turk (AMT)[2] or similar services to collect humour judgments for each caption (Shahaf et al., 2015; Radev et al., 2016), work on hashtag games can leverage built-in social media features such as likes or retweets to serve as humour judgments. Third, hashtag games enable researchers to explore humour in a social context by allowing access to an author's previous tweets as well as their social networks.

3 Data

The decentralized and transient nature of hashtag games presents a challenge to data collection. To alleviate this, we focus on hashtag games created by the Comedy Central show *@midnight* as part of their nightly "Hashtag Wars" segment. This ensures that each game has a sufficiently large number of active participants and provides a regular source of hashtag game prompts. In this work, we create a corpus of responses for four specific hashtags: #GentlerSongs, #OlympicSongs, #BoringBlockbusters, and #OceanMovies. These hashtags all occurred between April and August, 2016, and, as mentioned in Section 1, were chosen specifically for their formulaic nature.

3.1 Humour Judgments

Users on Twitter show their approval of a tweet through likes and retweets. Thus, we use these to infer humour judgments. More concretely, we compute, for each tweet, the sum of the number of likes and the number of retweets to act as funniness indicators. For each hashtag game, these sums, which we will refer to as the total likes, are compared to generate pairwise relative humour judgments, with the tweet that received more total likes being considered funnier than tweet with fewer.

In our dataset, total likes followed a Zipfian distribution with over 56% of all collected tweets obtaining zero total likes. To help reduce the effects of noise in the data as well as to ensure accuracy in our humour judgments, this paper only considers tweets which received at least seven total likes. Although Twitter

[1] http://contest.newyorker.com/

[2] https://mturk.com/

Hashtag	# of Tweets Collected	# of Tweet with $\geq$ 7 total likes	# of pairwise judgments (excluding ties)
#GentlerSongs	12,543	256	29,874
#OlympicSongs	8,778	460	100,175
#OceanMovies	12,189	327	49,638
#BoringBlockbusters	11,599	198	18,149
All	**45,109**	**1,241**	**197,836**

Table 2: Tweet counts and number of pairwise judgments by hashtag game

allows users to both like and retweet the same tweet, it does not provide an easy way to detect this. A threshold of seven total likes guarantees a tweet has been rated by at least four individuals. This helps to smooth out any unreliable judgments such as bots or misclicks and ensure a tweet has wide-spread humour appeal. This threshold resulted in 197,836 pairwise relative humour judgments, excluding ties, as shown in Table 2.

In general, liking or retweeting a tweet can be seen as an implicit approval, e.g. as a show of agreement, to save a tweet for future use, or as an act of curation (Boyd et al., 2010; Gorrell and Bontcheva, 2016). While it is easy to imagine scenarios where liking or retweeting is not an implicit approval, e.g. retweeting to provide context for a critique, at least in the case of hashtag games, such scenarios seem to be quite rare. In fact, e-commerce literature use retweets as "a measure of community interest" (Gilbert et al., 2013).

The act of retweeting is a complex phenomenon and is affected not only by linguistic but paralinguistic features such as URLs, hashtags, and mentions, as well as extra-linguistic factors such as number of followers (Suh et al., 2010). In order to control for these factors we omit all tweets containing URLs, photos, videos, hashtags (other than the relevant hashtag game prompt), or mentions (other than the *@midnight* account). This has the added benefit of ensuring that the humour of a tweet is indeed drawn from the tweet text itself rather than through a contrast between the text and a photo or news story.

Another potential shortcoming is that likes and retweets are not independent. More retweets mean a greater audience and thus potentially more likes. However, likes and retweet are both used to express appreciation of a tweet (Boyd et al., 2010; Gorrell and Bontcheva, 2016), and liking and retweeting are considered separate actions on Twitter. Some users may like a tweet without retweeting it while others may retweet without liking. Therefore, drawing humour judgments from only likes or only retweets would ignore a large portion of the available data. Furthermore, it would fail to capture scenarios where a user both likes and retweets the same tweet, which can be seen as an even stronger expression of appreciation than liking or retweeting alone. As mentioned above, since Twitter does not offer an easy way to tell when a user likes and retweets the same tweet, the easiest way to add weight such scenarios is through a simple sum.

As mentioned in Section 2, one potential alternative to using total likes as de facto humour judgments would be to collect gold standard pairwise humour judgments through AMT or similar service. While this may result in more trustworthy humour judgments, the collection process would be relatively time consuming and expensive. Furthermore, practical constraints may prevent researchers from obtaining pairwise judgments for all possible pairs. By comparison, like and retweet counts are built into the Twitter API[3] and require very little extra processing time or cost. Additionally, obtaining pairwise judgments for every possible pair is trivial. Although total likes is not a perfect metric for discerning humour, it still offers the easiest indication of how much users enjoyed a particular tweet. That said, an in-depth comparison of total likes versus gold standard humour judgments is a potential topic for future work.

[3]https://dev.twitter.com/

3.2 Punchline Annotation

It was necessary to first identify what the punchlines and setups in a tweet are in order to examine their semantic relatedness. As mentioned in Section 1, we focus on a specific type of hashtag game where well known quotes/lyrics/titles/etc. are taken from a source domain and modified with references to a target domain. Responses to #GentlerSongs and #OlympicSong tended to be variations on song titles or lyrics while responses to #OceanMovies and #BoringBlockbusters tended to be variations on movie titles.

A professional comedian and joke writer was invited to manually annotate the punchlines. Punchlines were loosely defined as the set of words which appear in a tweet that do not appear in original title/lyric, although the annotator was instructed to use their professional judgment in cases such as typos or minor misquotations. In fact, such situations were the reason we chose a human annotator over an automated approach involving partial text matching, although future works may explore this avenue. In cases where the annotator was unable to identify the original title/lyric, the tweet was omitted from the data. Setups were defined as the adjective part of the hashtag prompts, i.e. "gentler" for #GentlerSongs, "Olympic" for #OlympicSongs, etc.

4 Features

4.1 Measures of Semantic Relatedness

This paper considers five different measures of semantic relatedness. The first three measures are based on free word association (FWA) norms. Nelson et al. (1998) presented participants with a list of English words and instructed them "to write the first word that came to mind that was meaningfully related or strongly associated to the presented cue word." The proportion of respondents who produced word Y when presented with a cue word X is referred to as the forward strength from X to Y. It is important to note that forward strength is directional, i.e. participants may be more likely to produce "green" given the cue "grass" than to produce "grass" given the cue "green".

Due to the sparse nature of the FWA dataset, we define the *FWA strength* between two words as the product of the forward strengths along the shortest path between them. We compute this value by constructing a graph where each node U corresponds to a word in the Nelson et al. (1998) FWA norm vocabulary and each edge U, V has a weight proportional to $-log(f(U,V))$ where $f(U,V)$ is the forward strength between words U and V. The FWA strength is equal to $exp(cost(U,V))$ where $cost(U,V)$ is the cost of the shortest path from U to V according to Dijkstra's algorithm.

As we are interested in the semantic relationships between setups and punchline, we define $FWA_{forward}$ as the strength with which the setup conjures the punchline and $FWA_{backward}$ as the strength with which the punchline conjures the setup. Again, due to the directional nature of FWA, these values represent subtly different phenomena. We are also interested in how these measures interact so we define $FWA_{difference}$ as $\text{FWA}_{\text{forward}} - \text{FWA}_{\text{backward}}$.

The fourth measure is *Word2Vec similarity* (Mikolov et al., 2013), which we will simply refer to as Word2Vec. Word2Vec was trained using Gensim (Rehurek and Sojka, 2010) on English-language Wikipedia using a continuous bag-of-words model with feature vectors of dimensionality 400. Wikipedia was chosen as the training corpus in an attempt to capture world knowledge. We experimented with training Word2Vec on a 1,600,000 tweet corpus compiled in Go et al. (2009) but found it performed worse than Wikipedia, likely due to its relatively small sample size.

Finally, the fifth measure is the *Normalized Google Distance* (NGD) (Cilibrasi and Vitanyi, 2007). NGD represents the "normed semantic distance between the terms in question... in the cognitive space invoked by the usage of the terms on the world-wide-web as filtered by Google". In short, NGD offers an easy way to leverage not only the vast chunk of the word-wide-web indexed by Google but also the power of Google Search itself. Being a distance, NGD is unlike Word2Vec and FWA features in that smaller values represent stronger relationships.

We compute all measures between each tweet's setup and each word in the corresponding punchline, as defined in Section 3.2. We record the highest value pair, lowest value pair, and average value. It should be noted that specifically in the case of $\text{FWA}_{\text{difference}}$, $\text{FWA}_{\text{difference}}$ (highest) does not correspond to the

setup/punch word pair with the greatest difference between $FWA_{forward}$ and $FWA_{backward}$ but rather the difference between $FWA_{forward}$ (highest) and $FWA_{backward}$ (highest).

4.2 Perplexity and POS Perplexity

We calculate the tweet-level *perplexity* and *POS perplexity* to serve as a baseline. This follows Shahaf et al. (2015) which found perplexity and POS perplexity to be simple yet effective methods for identifying the funnier of a pair of cartoon captions. Due to the similarities between cartoon captions and hashtag game responses noted in Section 2, we expect that perplexity should also be useful in identifying funnier hashtag game responses. Perplexity was calculated using 2-gram, 3-gram, and 4-gram language models trained using KenLM (Heafield et al., 2013) on English-language Wikipedia. POS perplexity was trained in a similar way but with each word in the training corpus being replaced by its respective POS tag according to NLTK[4]. As with Word2Vec, we experimented with language models trained on the same Go et al. (2009) tweet corpus tagged using Tweet NLP (Gimpel et al., 2011) but found it performed worse than Wikipedia.

Shahaf et al. (2015) note that funnier cartoon captions tend to use "simpler grammatical structure", i.e. have a lower POS perplexity. Their results for perplexity were less clear. While a lower perplexity, i.e. "less-distinctive language", was preferred when comparing captions with similar punchlines, a higher perplexity was preferred when comparing captions with different punchlines.

5 Results and Discussion

The statistics for each feature are shown in Table 3. Following Shahaf et al. (2015), results are shown as the percentage of pairs for which the higher value belonged to the funnier tweet, i.e. the tweet with more total likes. Values above 50% imply a positive correlation between that feature and perceived funniness, values below 50% imply a negative correlation. Significance was calculated using a two-sided Wilcoxon signed rank test. Since we consider multiple features, Holm-Bonferroni correction was employed to reduce the chance of a Type-I error. Although the reported results are close to the expectation by chance, 50%, many features showed a high degree of significance. Furthermore, these results are similar in magnitude to the results reported in Shahaf et al. (2015).

The results show that perplexity is relatively effective in identifying funnier tweets. This is in line with both our expectations and with the results of Shahaf et al. (2015). However, while Shahaf et al. (2015) found that lower perplexity was funnier only when comparing cartoon captions with similar punchlines, hashtag game responses with lower perplexity tended to be judged as funnier regardless of the similarity between tweets' punchlines. This indicates a preference for simpler vocabulary, possibly because a simpler vocabulary allows punchlines to be more easily understood.

In agreement with Shahaf et al. (2015), funnier tweets also tended to have slightly lower POS perplexity, indicating simpler grammatical structures. The relatively slight effect of POS perplexity compared to Shahaf et al. (2015), as well as the improved performance of the 2-gram language model over 3-grams and 4-grams, may be due to differences between the training and test corpora. Wikipedia and Twitter use very different styles of language. Although we expect that training language models on tweets, or even song lyrics, movie quotes, etc., would improve performance, as mentioned in Section 4.2 this would require an appropriate corpus and is a topic for future work.

Although we expected weaker relationships between setups and punchlines to be less humourous, the overall trend across all semantic relatedness measures was a notable preference for punchlines which are less related to setups (higher NGD, lower Word2Vec and FWA features). This seems counterintuitive at first as one would reasonably expect low NGD, high Word2Vec, or high FWA strengths to be funnier. However, this is not the case. One possible explanation is that, since we expect punchlines should be unexpected, punchlines with too low an NGD, too high a Word2Vec similarity, or too high FWA strengths may be too obvious and thus less funny. This is illustrated in Figure 1a which shows that while lower $FWA_{backward}$ scores do not necessarily result in funnier tweets, funnier tweets tend to have lower $FWA_{backward}$ scores. This is also reinforced by the fact that Word2Vec and $FWA_{forward}$ were the

[4]http://www.nltk.org/

Feature		% Funnier is Higher
Perplexity	(2-gram)	47.82**
	(3-gram)	47.88**
	(4-gram)	47.86**
POS Perplexity	(2-gram)	49.18**
	(3-gram)	49.47**
	(4-gram)	49.46**
$FWA_{forward}$	(highest)	48.42**
	(lowest)	48.40**
	(average)	48.61**
$FWA_{backward}$	(highest)	49.47**
	(lowest)	49.52
	(average)	49.38
$FWA_{difference}$	(highest)	48.38
	(lowest)	48.53**
	(average)	47.41**
Word2Vec	(highest)	49.63
	(lowest)	48.98**
	(average)	49.15**
NGD	(highest)	52.45**
	(lowest)	50.57**
	(average)	51.69**

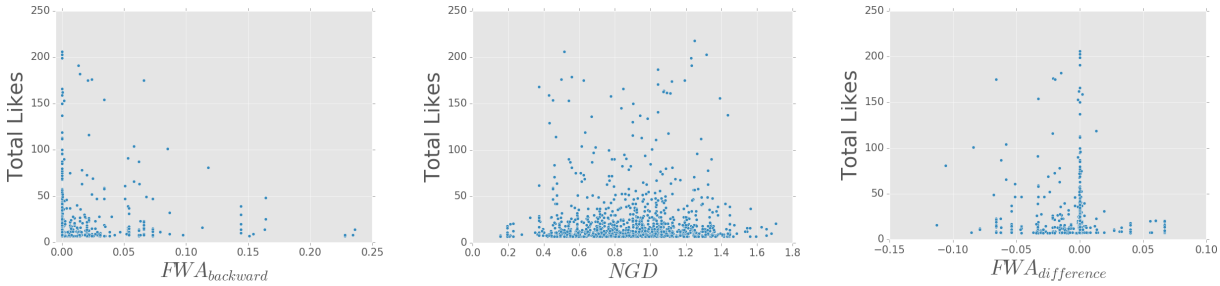
Table 3: Percentage of caption pairs where funnier tweet contains the higher feature value. Significance according to a two-sided Wilcoxon signed rank test is indicated using *-notation (*p$\leq$0.05, **p$\leq$0.005, Holm-Bonferroni correction)

most predictive when considering the lowest value (least similar/weakest) setup/punch word pairs, while NGD was the most predictive when considering the highest (most distant) setup/punch word pairs.

Following the intuition that punchlines should be related to setups but should also not be obvious, one would expect that as NGD increases or Word2Vec/$FWA_{forward}$/$FWA_{backward}$ decrease, funniness should drop off after a certain point. While Figure 1a shows that this is not the case for $FWA_{backward}$, Figure 1b does seem to suggest it is for NGD. It may be the case that funnier punchlines are as obscure as possible while still having some recognizable connection to their corresponding setups. This would also help explain the increase in variance as $FWA_{backward}$ approaches 0; the less obvious the relation between punchline and setup is, the higher the upper bound on funniness but the greater the likelihood of the punchline not being understood. If this is the case it is not surprising that Word2Vec or FWA features failed to capture the expected drop off, nor that NGD succeeded in doing so, as they are trained on relatively small corpora compared to the amount of pages indexed by Google. Another advantage of NGD is that since Google is constantly indexing new pages, including news sites, NGD is able to capture emerging topical relationships that fixed corpora cannot, such as the controversy surrounding the Zika virus outbreak in Brazil during the 2016 Rio Olympic Games.

While both NGD and Word2Vec are symmetric, FWA features are not. Following the intuitions that punchlines should be unexpected and that punchlines should have some relation to the setup, one would expect that punchlines with low $FWA_{forward}$ but high $FWA_{backward}$ would be deemed funnier. A relatively weak $FWA_{forward}$ would suggest the punchline is unexpected given the setup while a relatively strong $FWA_{backward}$ would suggest the relationship between the punchline and the setup is easily recognizable. In other words, a punchline should be difficult to think of yourself while easy to understand.

Not only does this intuition appear to be correct but $FWA_{difference}$ is more predictive than $FWA_{forward}$ or $FWA_{backward}$ alone. Although the funniest tweets had an $FWA_{difference}$ of near 0, Figure 1c clearly shows that tweets with a negative $FWA_{difference}$ have a much greater potential to be judged as funny compared



(a) (b) (c)

Figure 1: Total likes by (a) $\text{FWA}_{\text{backward}}$, (b) NGD, and (c) $\text{FWA}_{\text{difference}}$

to tweets with a positive one. However there is a trade-off between $\text{FWA}_{\text{difference}}$ and $\text{FWA}_{\text{backward}}$. As $\text{FWA}_{\text{difference}}$ becomes more negative, either $\text{FWA}_{\text{forward}}$ has to become smaller or $\text{FWA}_{\text{backward}}$ has to become larger. While decreasing $\text{FWA}_{\text{forward}}$ might actually increase funniness, the danger is that if $\text{FWA}_{\text{backward}}$ becomes too large then the tweet would become less funny.

One shortcoming of the FWA dataset is its relatively small vocabulary and sparse connectedness. For the hashtag #GentlerSongs, valid paths from the setup, "gentler", to at least one punch word were found in only 61.16% of tweets. Valid paths from some punch word to the setup occurred in only 49.72% of tweets. Only 21.03% of tweets had both. Obviously, this lack of coverage limits the widespread effectiveness of FWA features as well as the confidence with which we can view the results.

Finally, although we examine the highest, lowest, and average value per tweet for each of our five semantic measures, with the exception of NGD, all results were within a single percentage point of each other. This lack of variance can be at least partly attributed to the fact that punchlines in our dataset tended to be very short, averaging only 1.37 words per tweet.

6 Conclusions and Future Work

In this paper we explored the effects of semantic relatedness between setup and punchlines in Twitter hashtag games. To this end, we collected responses for four different hashtag games created by the Comedy Central show *@midnight* and used like/retweet counts to form pairwise relative humour judgments. We investigated five potential semantic relatedness measures and found perplexity, NGD, and $\text{FWA}_{\text{difference}}$ to be the most consistent indicators of funniness.

Additionally, we have provided empirical evidence of a preference against obvious jokes with funnier tweets tending to show weaker semantic relationships using symmetric measures of relatedness (NGD and Word2Vec). The asymmetric nature of the FWA features allows us to compare how easy it is to produce a punchline given only the setup versus how easy it is to recognize the connection between a punchline and a setup. Interestingly, we show that while punchlines should be easier to recognize than they are to produce, punchlines which are overall harder to recognize still tend to be judged as funnier.

Although this work represents only a first step towards a full humour recognition system, we believe semantic relatedness between setups and punchlines is worthy of further examination. Furthermore, we believe the task of humour recognition for Twitter hashtag games in general is an extremely promising area for computational humour research.

This paper focused on a relatively small subset of responses for only four different hashtag games, all relating to either songs or movies. Examining more tweets across a more diverse set of hashtag game prompts would allow for more easily generalized results. This work would be further improved by automatic punchline identification. The reliance on human punchline annotations prevents this work from being applied to a larger dataset. Additionally, while FWA feature results are promising, a lack of coverage means it is unlikely that FWA features will see wide spread use. However, they do suggest that asymmetrical measures of semantic relatedness deserve further examination.

In this work we defined the punchline as the deviation from the source domain (song titles or lyrics in the case of #GentlerSongs and #OlympicSongs; movie title in the case of #OceanMovies and #Boring-

Blockbusters). However, a tweet's humour does not come from such deviations alone. Quality of puns, multiple deviations, and even popularity of the source title/lyric can all affect perceived funniness. These features present obvious next steps for computational humour research into Twitter hashtag games. We expect their inclusion would not only improve results and but also lead to a more comprehensive model of hashtag game humour.

Finally, while this work focused on a specific type of hashtag game which tends to attract formulaic responses, hashtag games can be more complex. Word count related hashtags like #CollegeIn5Words, #MyLoveLifeIn3Words, etc. as well as open-ended hashtags like #WhenIWasYourAge, #WrongReasonsToHaveKids, etc. do not follow such formulas and thus present a significantly larger challenge to humour recognition. We intend to explore such hashtags in future works.

References

Binsted, K. & Ritchie, G. (1994) An implemented model of punning riddles. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (Vol. 1)*. AAAI.

Boyd, D., Golder, S., & Lotan, G. (2010, January). Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In *System Sciences (HICSS), 2010 43rd Hawaii International Conference on (pp. 1-10)*. IEEE.

Cilibrasi, R. L., & Vitanyi, P. M. (2007). The google similarity distance. In *IEEE Transactions on knowledge and data engineering, 19(3), 370-383.*

Davidov, D., Tsur, O., & Rappoport, A. (2010, July). Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the fourteenth conference on computational natural language learning (pp. 107-116)*. ACL.

Erkan, G., & Radev, D. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. In *Journal of Artificial Intelligence Research, 22, 457-479*.

Gilbert, E., Bakhshi, S., Chang, S., & Terveen, L. (2013, April). I need to try this?: a statistical overview of pinterest. In *Proceedings of the SIGCHI conference on human factors in computing systems (pp. 2427-2436)*. ACM.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, F., & Smith, N. A. (2011, June). Part-of-speech tagging for twitter: Annotation, features, and experiments In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2 (pp. 42-47)*. ACL.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision *CS224N Project Report, Stanford, 1, 12.*.

Gorrell, G., & Bontcheva, K. (2016). Classifying twitter favorites: like, bookmark, or thanks?. In *Journal of the Association for Information Science and Technology, 67(1), 17-25*.

Heafield, K., Pouzyrevsky, I., Clark, J. H., & Koehn, P. (2013, August) Scalable Modified Kneser-Ney Language Model Estimation. In *ACL (2) (pp. 690-696)*.

Labutov, I., & Lipson, H. (2012, July). Humor as circuits in semantic networks. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2 (pp. 150-155)*. ACL.

Mihalcea, R., & Pulman, S. (2007, February). Characterizing humour: An exploration of features in humorous texts. In *International Conference on Intelligent Text Processing and Computational Linguistics (pp. 337-347)*. Springer Berlin Heidelberg.

Mihalcea, R., & Strapparava, C. (2005, October). Making computers laugh: Investigations in automatic humor recognition. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (pp. 531-538)*. ACL.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*.

Morkes, J., Kernal, H. K., & Nass, C. (1998, April). Humor in task-oriented computer-mediated communication and human-computer interaction. In *CHI 98 Conference Summary on Human Factors in Computing Systems (pp. 215-216)*. ACM.

Nelson, D. L., McEvoy, C. L., & Schreiber, T. A. (1998). The University of South Florida word association, rhyme, and word fragment norms. http://www.usf.edu/FreeAssociation/.

Petrovic, S., & Matthews, D. (2013, August). Unsupervised joke generation from big data. In *ACL (2) (pp. 228-232)*.

Radev, D., Stent, A., Tetreault, J., Pappu, A., Iliakopoulou, A., Chanfreau, A., de Juan, P., Vallmitjana, J., Jaimes, A., Jha, R., & Mankoff, B. (2016). Humor in Collective Discourse: Unsupervised Funniness Detection in the New Yorker Cartoon Caption Contest. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. ELRA.

Rehurek, R., & Sojka, P. (2010) Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*.

Reyes, A., Rosso, P., & Buscaldi, D. (2012). From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering, 74, 1-12*.

Ritchie, G., Manurung, R., Pain, H., Waller, A., Black, R., & O'Mara, D. (2007). A practical application of computational humour. In *Proceedings of the 4th International Joint Conference on Computational Creativity (pp. 91-98)*.

Shahaf, D., Horvitz, E., & Mankoff, R. (2015, August). Inside jokes: Identifying humorous cartoon captions. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1065-1074)*. ACM.

Sheridan, Rob. (2011, September 15). The Enthusiast: Hashtag Games [Web log post]. Retrieved from http://6thfloor.blogs.nytimes.com/2011/09/15/the-enthusiast-hashtag-games/?_r=0

Stein, Scott. (2016, October 10). Google Assistant uses joke writers from Pixar and The Onion Retrieved from https://www.cnet.com/news/google-hired-pixar-and-onion-joke-writers-for-assistant/

Stock, O., & Strapparava, C. (2002). HAHAcronym: Humorous agents for humorous acronyms. *Stock, Oliviero, Carlo Strapparava, and Anton Nijholt. Eds, 125-135*.

Suh, B., Hong, L., Pirolli, P., & Chi, E. H. (2010, August). Want to be retweeted? large scale analytics on factors impacting retweet in twitter network. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on Social Computing (pp. 177-184)*. IEEE.

Taylor, J., & Mazlack, L. (2004, August). Computationally recognizing wordplay in jokes. In *Proceedings of CogSci (Vol. 2004)*.

Tsur, O., Davidov, D., & Rappoport, A. (2010, May). ICWSM-A Great Catchy Name: Semi-Supervised Recognition of Sarcastic Sentences in Online Product Reviews. In *ICWSM*.