

Parody Detection

An Annotation, Feature Construction, and Classification Approach to the Web of Parody

Joshua L. Weese¹, William H. Hsu¹, Jessica C. Murphy², and Kim Brillante Knight³

Abstract

In this chapter, we discuss the problem of how to discover when works in a social media site are related to one another by artistic appropriation, particularly parodies. The goal of this work is to discover concrete link information from text expressing how this may entail derivative relationships between works, authors, and topics. In the domain of music video parodies, this has general applicability to titles, lyrics, musical style, and content features, but the emphasis in this work is on descriptive text, comments, and quantitative features of songs. We first derive a classification task for discovering the "Web of Parody." Furthermore, we describe the problems of how to generate song/parody candidates, collect user annotations, and apply machine learning approaches comprising of feature analysis, construction, and selection for this classification task. Finally, we report results from applying this framework to data collected from *YouTube* and explore how the basic classification task relates to the general problem of reconstructing the web of parody and other networks of influence. This points toward further empirical study of how social media collections can statistically reflect derivative relationships and what can be understood about the propagation of concepts across texts that are deemed interrelated.

¹ Department of Computer Science, Kansas State University

² School of Arts and Humanities, University of Texas at Dallas

³ School of Arts, Technology, and Emerging Communication, University of Texas at Dallas

1. Toward a Web of Derivative Works

We consider the problem of reconstructing networks of influence in creative works –specifically, those consisting of sources, derivative works, and topics that are interrelated by relations that represent different modes of influence. In the domain of artistic appropriation, these include such relationships as “B is a parody of A”. Other examples of “derivative work” relationships include expanding a short story into a novel, novelization of a screenplay or the inverse (adapting a novel into a screenplay). Still more general forms of appropriation include quotations, mashups from one medium into another (*e.g.*, song videos), and artistic imitations. In general, *derivative work* refers to any expressive creation that includes major elements of an original, previously created (*underlying*) work.

The task studied in this paper is detection of source/parody pairs among pairs of candidate videos on *YouTube*, where the parody is a derivative work of the source. Classifying an arbitrary pair of candidate videos as a source and its parody is a straightforward task for a human annotator, given a concrete and sufficiently detailed specification of the criteria for being a parody. However, solving the same problem by automated analysis of content is much more challenging, due to the complexity of finding applicable features. These are multimodal in origin (*i.e.*, may come from the video, audio, metadata, comments, etc.); admit a combinatorially large number of feature extraction mechanisms, some of which have an unrestricted range of parameters; and may be irrelevant, necessitating some feature selection criteria.

Our preliminary work shows that by analyzing only video information and statistics, identifying correct source/parody pairs can be done with an ROC area of 65-75%. This can be improved by doing analysis directly on the video itself, such as Fourier analysis and extraction of lyrics (from closed captioning, or from audio when this is not available). However, this analysis is computationally intensive and introduces error at every stage. Other information can be gained by studying the social aspect of *YouTube*, particularly how users interact by commenting on videos. By introducing social responses to videos, we are able to identify source/parody pairs with an f-measure upwards to 93%.

The novel contribution of this research is that, to our knowledge, parody detection has not been applied in the *YouTube* domain, nor by analyzing user comments. The central hypothesis of this study is that by extracting features from *YouTube* comments, performance in identifying correct source/parody pairs will improve over using only information about the video itself. Our experimental approach is to gather source/parody pairs from *YouTube*, annotating the data, and constructing features using analytical component libraries, especially natural language toolkits. This demonstrates the feasibility of detecting source/parody video pairs from enumerated candidates.

1.1 Context: Digital Humanities and Derivative Works

The framing contexts for the problem of parody detection are the *web of influence* as defined by Koller (2001): graph-based models of relationships, particularly first-order relational extensions of probabilistic

graphical models that include a representation for universal quantification. In the domain of digital humanities, a *network of influence* consists of creative works, authors, and topics that are interrelated by relations that represent different modes of influence. The term “creative works” includes texts and also products of other creative domains, and include musical compositions and videos as discussed in this chapter. In the domain of artistic appropriation, these include such relationships as “B is a parody of A”. Other examples of “derivative work” relationships include expanding a short story into a novel, novelization of a screenplay or the inverse (adapting a novel into a screenplay). Still more general forms of appropriation include quotations, mashups from one medium into another (*e.g.*, song videos), and artistic imitations.

The technical objectives of this line of research are to establish representations for learning and reasoning about the following tasks:

1. how to discover when works are related to one another by artistic appropriation
2. how this may entail relationships between works, authors, and topics
3. how large collections (including text corpora) can statistically reflect these relationships
4. what can be understood about the propagation of concepts across works that are deemed interrelated

The above open-ended questions in the humanities pose the following methodological research challenges in informatics: specifically, how to use machine learning, information extraction, data science, and visualization techniques to reveal the network of influence for a text collection.

1. **(Problem)** How can relationships between documents be detected? For example, does one document extend another in the sense of textual entailment? If statement A *extends* statement B, then B entails A. For example, if A is the assertion “F is a flower” and B is the assertion “F is a rose”, then A extends B. Such extension (or appropriation) relationships serve as building blocks for constructing a web of influence.
2. **(Problem)** What entities and features of text are relevant to the extension relationship, and which of these features transfer to other domains?
3. **(Technology)** What are algorithms that support relationship extraction from text and how do these fit into information extraction (IE) tools for reconstructing entity-relational models of documents, authors, and inspirational topics?
4. **(Technology)** How can information extraction be integrated with search tasks in the domain of derivative works? How can creative works, and their supporting data and metadata, support free-text user queries in portals for accessing collections of these works?
5. **(Technology)** How can newly-captured relationships be incorporated and accounted for using ontologies and systems of reasoning that can capture semantic entailment in the above domain.
6. **(System)** How can a system be developed that maps out the spatiotemporal trajectory of an entity from the web of influence? For example, how can the propagation of an epithet, meme, or individual writing style from a domain of origin (geographic, time-dependent, or memetic) be visualized?

The central thesis of this work is that this combined approach will enable link identification towards discovering networks of influence in the digital humanities, such as among song parody videos and their authors and original songs. The need for such information extraction tools arises from the following present issues in text analytics for relationship extraction, which we seek to generalize beyond text. System components are needed for:

1. expanding the set of known entities
2. predicting the existence of a link between two entities
3. inferring which of two similar works is primary and which is derivative
4. classifying relationships by type
5. identifying features and examples that are relevant to a specified relationship extraction task

These are general challenges for information extraction, not limited to the domain of modern English text, contemporary media studies, or even digital humanities.

1.2 Problem Statement: The Web of Parody

Goal: To automatically analyze the metadata and comments of music videos on a social video site (*YouTube*) and extract features to develop a machine learning-based classification system that can identify source/parody music videos from a set of arbitrary pairs of candidates.

The metadata we collected consists of quantitative features (descriptive statistics of videos, such as playing time) and natural language features. In addition to this metadata, the video contents can be analyzed using acoustical analysis to recognize sung lyrics (Mesaros & Virtanen, 2010) or image recognition to recognize human actions in music videos (Liu, Ali, & Shah, 2008). Such sophisticated multimedia processing is, however, computationally intensive, meaning data analysis takes orders of magnitude longer and requires sophisticated hardware. Moreover, while the residual error is in generally excess of 25%, the potential reduction using natural language features is hypothesized to be significant. As our experiments show, this is indeed the case, using topic modeling features derived from descriptor text and comments had far lower computational costs than those of extracting audiovisual features from video. The remaining residual error makes any achievable marginal improvement from multimedia analyses too small to be cost effective, and so we deem them to be beyond the scope of this work.

1.3 The Need for Natural Language Features

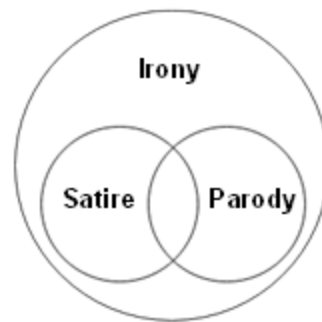


Figure 1. Literary devices used in derivative works.

The relative tractability of natural language analyses makes the language of derivative works the focus of this research. More importantly, we narrow the scope to discover whether the social response to a derivative work reflects its unique linguistic features. Derivative works employ different literary devices, such as irony, satire, and parody. As seen in Figure 1, irony, satire, and parody are interrelated. Irony can be described as appearance versus reality. In other words, the intended meaning is different the actual definition of the words (LiteraryDevices Editor 2014). For example, the sentence “We named our new Great Dane ‘Tiny’” is ironic because Great Dane dogs are quite large. Satire is generally used to expose and criticize weakness, foolishness, corruption, etc. of a work, individual, or society by using irony, exaggeration, or ridicule. Parody has the core concepts of satire; however, parodies are direct imitations of a particular work, usually to produce a comic effect.

2. Background and Related Work: Detecting Appropriated Works

2.1 Irony, Satire, and Parody Detection

Detecting derivative works can be a technically challenging task and is relatively novel beyond the older problems of plagiarism detection and authorship attribution. Baldwin and Burfoot (2009) introduce methodology in classifying satirical news articles as being either true (the real or original news article) or satirical. In a variety of cases, satire can be subtle and difficult to detect. Features focused on were mainly lexical, for example, the use of profanity and slang and similarity in article titles. In most cases, the headlines are good indications of satires, but so are profanity and slang since satires are meant for ridicule. Semantic validity was also introduced by using named entity recognition. This refers to detecting whether or not a named entity is out of place or used in the correct context.

Similar features can also be found in parodies. Bull (2010) focused on an empirical analysis on non-serious news, which includes sarcastic and parody news articles. Semantic validity was studied by

calculating the edit distance of common sayings. This expands beyond just parody as many writings use “common phrases with new spins”. Unusual juxtapositions and out of place language was also shown to be common in parody text, for example, “Pedophile of the Year” is phrase that is not uttered often in a serious context. This also leads to a comparison of the type of language used in parody and satirical articles. Non-serious text tends to use informal language with frequent use of adjectives, adverbs, contractions, slang and profanity, where serious text has more professional qualities of style, diction, tone, and voice. In contrast to serious text, parodies can also be personalized (use of personal pronouns). Punctuation was also seen as an indicator as serious text rarely use punctuation like exclamation marks. (Tsur, Davidov, & Rappoport 2010; Bull 2010).

As seen in **Error! Reference source not found.**, irony encompasses both satire and parody, but can also be more problematic to detect without a tonal reference or situational awareness. It is “unrealistic to seek a computational silver bullet for irony” (Reyes, Rosso, & Veale 2012). In an effort to detect verbal irony in text, Reyes, Rosso, & Veale (2012) focus on four main properties: signatures (typographical elements), unexpectedness, style (textual sequences), and emotional scenarios. Properties of irony detection clearly cascade down to the sub domains of parody and satire.

2.2 Music Video Domain

2.2.1 YouTube as a Data Source

YouTube has become one of the most popular user driven-video sharing platforms on the Web. In a study on the impact of social network structure on content propagation, Yoganarasimhan (2012) measured how *YouTube* propagated based on the social network a video was connected to (i.e. subscribers). He shed light on the traffic *YouTube* receives such that “In April 2010 alone, *YouTube* received 97 million unique visitors and streamed 4.9 billion videos” (Yoganarasimhan, 2012). Per recent reports from the popular video streaming service, *YouTube*’s traffic and content has exploded. *YouTube*, in 2016, had over a billion users, streamed hundreds of millions of hours of video each day, and spanned over 88 countries (Google, 2016). *YouTube* videos are also finding their way to social sites like *Facebook* (500 years of *YouTube* video watched every day) and *Twitter* (over 700 *YouTube* videos shared each minute). This leads to many research opportunities such as the goal of reconstructing a web of derivative works. With over 100 million people that like/dislike, favorite, rate, comment, and share *YouTube* videos, which makes *YouTube* a perfect platform to study social networks and relations.

2.2.2 The YouTube Social Network

YouTube is a large, content driven social network, interfacing with many social networking giants like *Facebook* and *Twitter* (Wattenhofer, Wattenhofer, & Zhu 2012). Considering the size of the *YouTube* network, there are numerous research areas, such as content propagation, virality, sentiment analysis, and content tagging. Recently, Google published work on classifying *YouTube* channels based on *Freebase* topics (Simmonet 2013). Their classification system worked on mapping *Freebase* topics to various categories for the *YouTube* channel browser. Other works focus on categorizing videos with a

series of tags using computer vision (Yang & Toderici 2011). However, analyzing video content can be computationally intensive.

To expand from classifying content based on content, this study looks at classifying *YouTube* videos based from social aspects like user comments. Wattenhofer, Wattenhofer, & Zhu (2012) performed large scale experiments on the *YouTube* social network to study popularity in *YouTube*, how users interact, and how *YouTube*'s social network relates to other social networks. By looking at user comments, subscriptions, ratings, and other related features, they found that *YouTube* differs from other social networks in terms of user interaction (Wattenhofer, Wattenhofer, & Zhu 2012). This shows that methodology in analyzing social networks such as *Twitter* may not be directly transferable to the *YouTube* platform. Diving further into the *YouTube* social network, Siersdorfer (2010) studied the community acceptance of user comments by looking at comment ratings and sentiment (Murphy, Hsu, Elshamy, Kallumadi, & Volkova, 2014; Trindade, Wang, Blackburn, & Taylor, 2014). Further analysis of user comments can be made over the life of the video by discovering polarity trends (Krishna, Zambreno, & Krishnan 2013).

2.3 Machine Learning Task: Classification

Machine learning, the problem of improving problem solving ability at a specified task given some experience (Mitchell, 1997), is divided by practitioners into several broad categories: *supervised learning*, which involves data for which a target prediction or classification is already provided by past observation or by a human annotator, and *unsupervised learning*, where the aim is to formulate categories or descriptors based on measures of similarity between objects, and these categories are not provided as part of input data. (Mitchell, 1997; Murphy, 2012; Alpaydin, 2014) Classifying previously unseen items based on known categories by training them on labeled texts is an instance of supervised learning (Mitchell, 1997), while *topic modeling*, the problem of forming as-yet unnamed categories by comparing members of a collection of items based on their similarities and differences, is a typical application of unsupervised learning (McCallum, 2002; Blei & Ng, 2003; Elshamy & Hsu, 2014). In text analytics, the items are text documents; however, we seek in this work and future work to extend the items being classified and categorized as derivative of others. That is, we seek to generalize to a broader range of creative works, including musical instruments or singers (Weese, 2014), musical compositions, videos, viral images and other memes, social media posts, users, and communities (Yang, Hsu, & Kallumadi, 2014), etc.

Over the last decade, researchers have focused on the use of the formulation of kernel-based methods with the purpose of determining similarity and indexing documents for such machine learning tasks as classification (Trindade, Wang, Blackburn, & Rooney, 2011) and clustering (Bloehdorn & Moschitti, 2007). The use of kernels allows a complex data space to be mapped to a compact feature space, where the level of similarity between documents can be easily and efficiently calculated using dynamic programming methods based on a kernel function (Doddington, et al., 2004; Shawe-Taylor & Cristianini, 2004). Such a kernel function forms the basis to a kernel machine such as support vector machine or online perception that can be applied for classification. The approach has been demonstrated to be

effective for various representations of documents in NLP from sequence kernels for POS tagging (Bunescu & Mooney, 2005; Lodhi, Saunders, Shawe-Taylor, Cristianini, & Watkins, 2002) to tree kernels based on parse trees (Cancedda, Gaussier, Goutte, & Renders, 2003). Moschitti has explored on the use of kernels for a number a specialized NLP tasks such as relation extraction (Nguyen, Moschitti, & Riccardi, 2009), semantic role labelling (Moschitti, 2006; Moschitti, Pighin, & Basili, 2008) and question and answer classification (Moschitti, 2008).

Relation extraction (RE), as defined by the Automatic Context Extraction (ACE) evaluation (Doddington, et al., 2004), is the task of finding semantic relations between pairs of named entities in text, e.g., organization, location, part, role, etc. ACE systems use a wide range of lexical, syntactic and semantic features to determine the relation mention between two entities. Supervised, semi-supervised and unsupervised machine learning methods have been applied to relation extraction. Supervised methods are generally the most accurate however with the proviso that there are only few relationship identified types and the corpus is domain-specific (Mintz, Bills, Snow, & Jurafsky, 2009). There has been extensive work in the latter direction with regard to the use of kernel methods. A number of kernel based approaches have been derived either through the use of one or more the following structural representations for a sentence: its constituent parse tree and its dependency based representation which encodes the grammatical dependencies between words. The approach of kernels over parse trees was pioneered by Collins and Duffy (2002), where the kernel function counts the number of common subtrees with appropriate weighting as the measure of similarity between two parse trees. Zelenko *et al.* (Zelenko, Aone, & Richardella, 2003) considered such use of parse trees for the purpose of relation extraction. Culotta and Sorensen (2004) extended this work to consider kernels between augmented dependency trees. Zhang et al. (2006) proposed the use of convolution kernels which provide a recursive definition over the structure (Moschitti, 2004). Nyugen et al. (2009) consider the use of a novel composite convolution kernels not just based on constituent parse trees but also for dependency and sequential structure for RE. A relation is represented by using the path-enclosed tree (which is the smallest sub-tree containing both entities) of the constituent parse tree or the path linking two entities of the dependency tree. Bunescu and Mooney (2005) proposed shortest path dependency kernel by stipulating that the only information to model a relationship between two entities can be captured by the shortest path between them in the dependency structure. The latter is represented as a form of subsequence kernel (Doddington, et al., 2004). Wang (2008) evaluated the latter structure in comparison to other subsequence kernels.

Kernels have been applied not only for relation extraction between named entities but more complex relationship learning discovery tasks between whole sentences such as question and answering and textual entailment. Moschitti et al. (2008) propose a kernel mechanism for text fragment similarity based on the syntactic parse trees.

3. Methodology: Using Machine Learning to Detect Parody

3.1 Feature Analysis and Selection

We treat the problem of parody detection over candidate source-video pairs as a classification task given computable ground features. Similar task definitions are used for prediction of friends in social networks: e.g., classification of a proposed direct friendship link as extant or not (Hsu, Lancaster, Paradesi, & Weninger, 2007; Caragea, Bahirwani, Aljandal, & Hsu, 2009). This supervised inductive learning thus presents a simultaneous feature analysis (extraction) and selection task.

Finding quantitative ground features is in many instances a straightforward matter of interrogating the *YouTube* data model (API Overview Guide, 2014) to extract fields of interest. In some social media analytics domains, this produces attributes that are irrelevant to some inductive learning algorithms (Hsu, Lancaster, Paradesi, & Weninger, 2007); in this domain, however, we found the effects of feature selection wrappers to be relatively negligible. By contrast, natural language features generally require crawling and parsing free text to extract sentiment, keywords of interest (including suppressed stopwords) and ultimately named entities.

3.2 Annotation for Supervised Learning

Ground truth for the supervised learning task is obtained by developing a user interface that presents candidate pairs of videos to an annotator, renders the metadata as it appears in *YouTube*, allows the annotator to view the video, and having him or her provide a Boolean-valued judgment as to whether the pair consists of a source and parody. No special expertise is required; no explanations are elicited; and this approach admits validation via annotator agreement *cf.* (Hovy & Lavid, 2010).

3.3 Addressing the Class Imbalance Problem

Class imbalance occurs when there is a significantly large number of examples of a certain class (such as positive or negative) over another. Drummond and Holte (2012) discuss the class imbalance problem as cost in misclassification. As the imbalance increases, algorithms like Naïve Bayes that are somewhat resistant to the class imbalance problem suffer performance. Instead of using different algorithms to overcome class imbalance, the authors suggest generalizing the data to create a more uniform distribution to help overcome class imbalance. There are various methods to create a more uniform distribution of classes in a data set. *YouTube* has millions of videos with a fraction of those being source/parody pairs. In order to keep the dataset in this study from becoming imbalanced, candidate source/parody pairs were filtered to give improved representation.

4. Data Acquisition and Preparation

4.1 Data Collection and Preprocessing

4.1.1 Criteria for Generation of Candidates

One challenge to overcome was that there is no parody dataset for *YouTube* and no concrete way of collecting such data. Our initial dataset included only information about the *YouTube* video (video statistics), rather than the video itself. The search for videos was quite limited (search bias in which videos were chosen). Given a well-known or popular parody video, the corresponding known source was found. The problem of multiple renditions of the same source arose and to solve it, only deemed “official” sources were collected (another search bias). The term “official” refers to the video being published (uploaded) by the artistic work’s artist or sponsor *YouTube* channel or account. The collection of known sources and parodies (28 of each) were retrieved using Google’s *YouTube* API and stored into an XML file format for easy access.

The final experimentation greatly expanded the preliminary dataset. Kimono Labs, an API for generating crawling templates, was used to generate seeds for crawling *YouTube* for source and parody videos (Kimono Labs 2014). The Kimono API allowed quick and easy access to the top 100 songs from *billboard.com* (the week of November 3rd was used). The song titles were collected and used to retrieve the top two hits from *YouTube* using the *YouTube* Data API (API Overview Guide 2014). Parodies were retrieved in a similar fashion, except the keyword “parody” was added to the *YouTube* query which was limited to the top five search results. This helped reduce the class imbalance problem. Pairs were generated by taking the cross product of the two source videos and the five parody videos, making 1474 videos after filtering invalid videos and videos that were not in English. The cross product was used to generate candidate pairs since source videos spawn multiple parodies as well as other fan made source videos. Information retrieved with the videos included the video statistics (view count, likes, etc.) and up to 2000 comments.

4.1.2 Annotation

A custom annotator was built to allow users to label candidate source/parody pairs as valid or invalid. This was a crucial step in removing pairs that were not true parodies (false positive hits in the *YouTube* search results) of source videos. Naively, videos could be tagged based on whether the candidate parody video title contains parody keywords like “parody” or “spoof,” but this generates several incorrect matches with sources. Likewise, if a parody video is popular enough, it also appears in the search results for the corresponding source video. It is also important to note that source lyric videos and other fan made videos were included in the dataset, so as to extend preliminary data beyond “official” videos. Having only two annotators available, pairs that were marked as valid by both annotators were considered to be valid source/parody pairs. In future works, more annotators will be needed and as such, inter-annotator agreement can be verified by kappa statistics and other means. Annotation left only 571 valid pairs (38.74%), which shows the importance of annotating the data versus taking the naïve approach to class labels. The number of pairs used in the final dataset were reduced to 162 valid pairs (about 11%) and 353 invalid pairs (23.95%) after removing videos that did not have a minimum of 100 comments available for crawling.

4.2 Feature Analysis

Preliminary experiments included four different feature sets:

1. The first used only ratios of video statistics (rating, number of times favorited, number of likes/dislikes, etc.) between the candidate source and parody.
2. The second used video statistic ratios plus a feature which indicated whether or not the second video in the pair was published after the first.
3. The third experiment used only the raw data collected (no ratios) plus the “published after” feature; this experiment was used as the baseline and used for comparison.
4. The fourth experiment included all features from the first three experiment designs.

The best performance was achieved as a result of the fourth feature set. The data set was also oversampled to reduce the class imbalance. This gave a 98% ROC area; however, using the raw data as features, along with the oversampling caused overfitting. A better representative of the preliminary results was an average ROC area of 65%-75%. Note that this is only with features generated from the video statistics.

4.3 Feature Extraction from Text

Table 1. Features of the final experiment. Note that each are unique to the source video and the parody video except TitleSimilarity which is for both.

Feature	Description
SentenceCount	Number of sentences from comments
Stanford NLP Sentiment	Sentiment of comment sentences which range from very negative to very positive (a five value system).
AvgCommentSentiment	Average word sentiment from TwitIE
BadWordCount	Percentage of words that are profanity
Penn Treebank NLP	The parts of speech tags in the Penn Treebank (Lieberman, 2003) as generated by Stanford NLP
Penn Treebank TwitIE	The parts of speech tags in the Penn Treebank generated by TwitIE
Punctuation	Punctuation marks
WordCount	Number of words in comments
AverageWordLength	Average length of words in comments
Top 20 Mallet Topics	The top 20 topics generated by Mallet for source videos and for parody videos.
Views	Number of views the video received
Likes	Number of likes for the video
Dislikes	Number of dislikes for the video
FavCount	Number of times the video was favorited
CommentCount	Number of comments the video has
TitleSimilarity	The edit distance of the parody and source video titles.

Extracting features from video content can come with a high computational overhead. Even though some natural language processing (NLP) tasks can be costly (depending on the size of text), this study focuses on using only features extracted from video information, statistics, and comments as shown in

Table 1. One area of focus were lexical features extracted from user comments per video. Parts of speech tags were generated by two different toolkits: Stanford NLP (Manning, et al. 2014) and GATE's TwitIE (Bontcheva, et al. 2013). This allows the evaluation of a short-text tagger (TwitIE) and a multipurpose tagger (Stanford NLP). Both were also used to analyze sentiment of user comments. TwitIE was used to produce an average word sentiment, where Stanford NLP was used for sentence level sentiment. Other features include statistical lexical and structural features like punctuation, average word length, and number of sentences. A profanity filter was used to calculate the number of bad words in each set of comments. The number of unrecognizable tokens by the parts of speech taggers was also added as a feature. This hints at the unique language of the user comments where nontraditional English spelling and internet slang is used. All counts (sentiment, parts of speech, etc.) were normalized to percentages to take into account the difference in the number of comments available between videos. Another large portion of features generated were by using Mallet (McCallum 2002), a machine learning toolkit for natural language. The built in stop word removal and stemming was used before collecting the top 20 topics for all parodies and sources for each training dataset. The summary of the process described in this section can be seen in Figure 2.

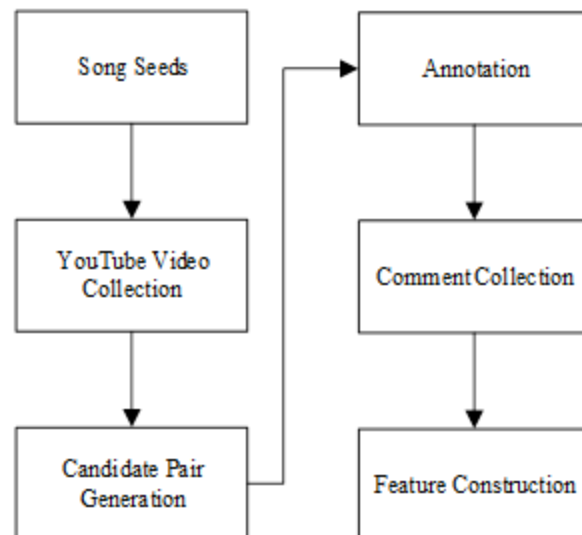


Figure 2. Workflow model of a system for collecting and classifying YouTube video source/parody pairs.

5. Experimental Results

5.1 Statistical Validation Approach

Experiments were conducted using a 10 fold cross validation with 90% of the data used for training and 10% used for testing. All features were generated per video automatically with the exception of a few features like title similarity, which requires both videos to construct the feature. Topic features were constructed by training the topic model in Mallet using the training datasets, and then using that model

to infer the topics for the test datasets. Two data configurations were used to test whether or not the occurrence of the word “parody” would introduce a bias to classification. A synset was created for removing these occurrences: {parody, parodies, spoof, spoofs}. The data configurations were then combined with different feature arrangements to test the impact of using Stanford NLP, TwitIE, and video statistics.

5.2 Results using Different Feature Sets

This section describes results on the parody-or-not classification task: learning the concept of a parody/original song pair by classifying a candidate pair (*Song1*, *Song2*) as being a parody paired with the original song it is based on. All classification tasks were done using the machine learning tool WEKA (Hall, et al. 2009). The supervised inductive learning algorithms (inducers) used included: Naïve Bayesian (NaiveBayes), instance-based (IB1), rule-based (JRip), decision tree (J48), artificial neural network (MLP), and logistic regression (Logistic).

Table 2. Results for the Stanford NLP, TwitIE, and video statistics feature set that include parody synsets.

Average F-Measure: Stanford NLP, TwitIE, and Video Statistics With Parody Synsets			
Inducers	AVG F Measure	STD	STD-ERR
IB1	91.01%	2.87%	0.91%
J48	90.78%	4.11%	1.30%
JRip	86.60%	5.06%	1.60%
Logistic	87.95%	2.87%	0.91%
MLP	91.35%	3.36%	1.06%
NaiveBayes	82.37%	4.21%	1.33%

Table 3. Results for the Stanford NLP, TwitIE, and video statistics feature set that exclude parody synsets.

Average F-Measure: Stanford NLP, TwitIE, and Video Statistics Without Parody Synset			
Inducer	AVG F Measure	STD	STD-ERR
IB1	91.39%	3.17%	1.00%
J48	85.67%	5.99%	1.89%
JRip	82.14%	5.58%	1.76%
Logistic	88.29%	3.72%	1.18%
MLP	90.44%	2.64%	0.83%
NaiveBayes	80.18%	3.87%	1.22%

Table 4. Results for the Stanford NLP and video statistics feature set that include parody synsets.

Average F-Measure: Stanford NLP and Video Statistics With Parody Synset			
Inducer	AVG F Measure	STD	STD-ERR
IB1	92.59%	2.96%	0.94%
J48	90.70%	4.21%	1.33%
JRip	85.80%	4.57%	1.45%
Logistic	86.80%	4.65%	1.47%
MLP	90.28%	3.65%	1.16%
NaiveBayes	82.55%	4.50%	1.42%

Table 5. Results for the Stanford NLP and video statistics feature set that exclude parody synsets.

Average F-Measure: Stanford NLP and Video Statistics Without Parody Synset			
Inducer	AVG F Measure	STD	STD-ERR
IB1	93.15%	3.19%	1.01%
J48	85.05%	4.88%	1.54%
JRip	86.16%	4.96%	1.57%
Logistic	81.73%	5.10%	1.61%
MLP	90.09%	3.06%	0.97%
NaiveBayes	78.50%	3.69%	1.17%

Table 6. Results for the Stanford NLP feature set that include parody synsets.

Average F-Measure: Stanford NLP With Parody Synset			
Inducer	AVG F Measure	STD	STD-ERR
IB1	92.39%	2.87%	0.91%
J48	88.50%	3.41%	1.08%
JRip	84.06%	3.83%	1.21%
Logistic	82.94%	6.41%	2.03%
MLP	88.73%	2.87%	0.91%
NaiveBayes	78.51%	4.70%	1.48%

Table 7. Results for the Stanford NLP feature set that exclude parody synsets.

Average F-Measure: Stanford NLP Without Parody Synset			
Inducer	AVG F Measure	STD	STD-ERR
IB1	92.94%	3.19%	1.01%
J48	86.87%	4.25%	1.34%
JRip	81.63%	4.80%	1.52%
Logistic	81.73%	5.03%	1.59%
MLP	87.71%	3.08%	0.97%
NaiveBayes	75.14%	4.11%	1.30%

Results were averaged across all 10 folds. The f-measure (Powers, 2011), standard deviation, and standard error can be found for each feature configuration in Table 2 through Table 7. On average, the best performing inducers were MLP and IB1 at 90%-93% f-measure. J48 performed well, but after looking at the pruned tree, the model tended to overfit. With the addition of features from user comments, performance increased significantly when compared to the preliminary work which used only video statistics. Stanford NLP (Table 4 and Table 5) is shown to overall produce more relevant features than the TwitIE parts of speech tagger (Table 2 and Table 3). When the TwitIE features were removed, performance was relatively unaffected (1%-2% at most). Logistic is an exception to this analysis as it dropped 6.59%; however, this is taken as an intrinsic property of the inducer and requires further investigation. The removal of the video statistic features, however, did reduce performance for most inducers, showing that the popularity of a video helps indicate the relation between a parody and

its source. Removing the parody synset did not have a heavy impact on performance. This is an important find, such that the word “parody” does not degrade classification of source/parody pairs.

5.3 Interpretation of Results: Topic and Feature Analysis

The most influential features were seen by using feature subset selection within *WEKA*. This showed that source and parody topics were most influential in the classification task. However, some topics clusters tend to overfit to popular videos or artist, especially for source videos. Generic clusters were also formed for things like music, humor, appraisal (users liked the song), and hate. A few unexpected topics also appeared, which show that current events also make it into the trending topics of the videos, for example: Obama, Ebola, and religion. Other feature analysis concluded that personal nouns were not relevant. This contradicts related work. Lexical features that were relevant included verbs, symbols, periods, adjectives, average word length in parody comments, and undefined or unrecognized tokens. Sentiment also showed promise during feature selection, though further experiments and dataset expansion will be needed to achieve more insightful feature selection.

The original hypothesis of this study is supported by the results. After introducing features extracted from comments, classification of source/parody pairs improved. The hypothesis also held after removing the parody synset. This generalizes the approach and makes it applicable to other domains, such as improving search, classifying news articles, plagiarism, and other derivative work domains. The proof of concept in this study leaves many possible directions for future research, including domain adaptation and feature expansion. Features left for future work include named entity recognition (this can help detect original authors of works), unusual juxtapositions and out of place language (Bull, 2010), sentence structure beyond punctuation (Reyes, Rosso, & Veale, 2012), and community acceptance of comments to supplement sentiment analysis (Siersdorfer, 2010).

6. Summary and Future Work

The results reported in Table 2 through Table 7 of this paper support the original hypothesis of this study: after introducing features extracted from comments, classification of source/parody pairs improved. More significantly, results obtained with the parody synset removed also support the hypothesis. This generalizes the approach and makes it applicable to other domains, such as improving search, classifying news articles, plagiarism, and other derivative work domains. The proof of concept in this study leaves many possible directions for future research, including domain adaptation, feature expansion, and community detection. Features left for future work include named entity recognition (this can help detect original authors of works), unusual juxtapositions and out of place language (Bull 2010), sentence structure beyond punctuation (Reyes, Rosso, & Veale 2012), and community acceptance of comments to supplement sentiment analysis (Siersdorfer 2010).

As mentioned in the introduction, a central goal of this work is to develop techniques and representations for heterogeneous information network analysis (HINA) to better support the discovery of webs of influence in derivation of creative works and the recognition of these and other instances of cultural appropriation. Figure 3 illustrates one such use case using early modern English ballads from the English Broadside Ballad Archive (EBBA); Figure 4 illustrates another based on the meme *Sí, se puede* (“Yes, one can”, popularly rendered “Yes, we can”). These are hand-constructed examples of the types of “network of influence” diagrams that we aim to produce in continuing research.

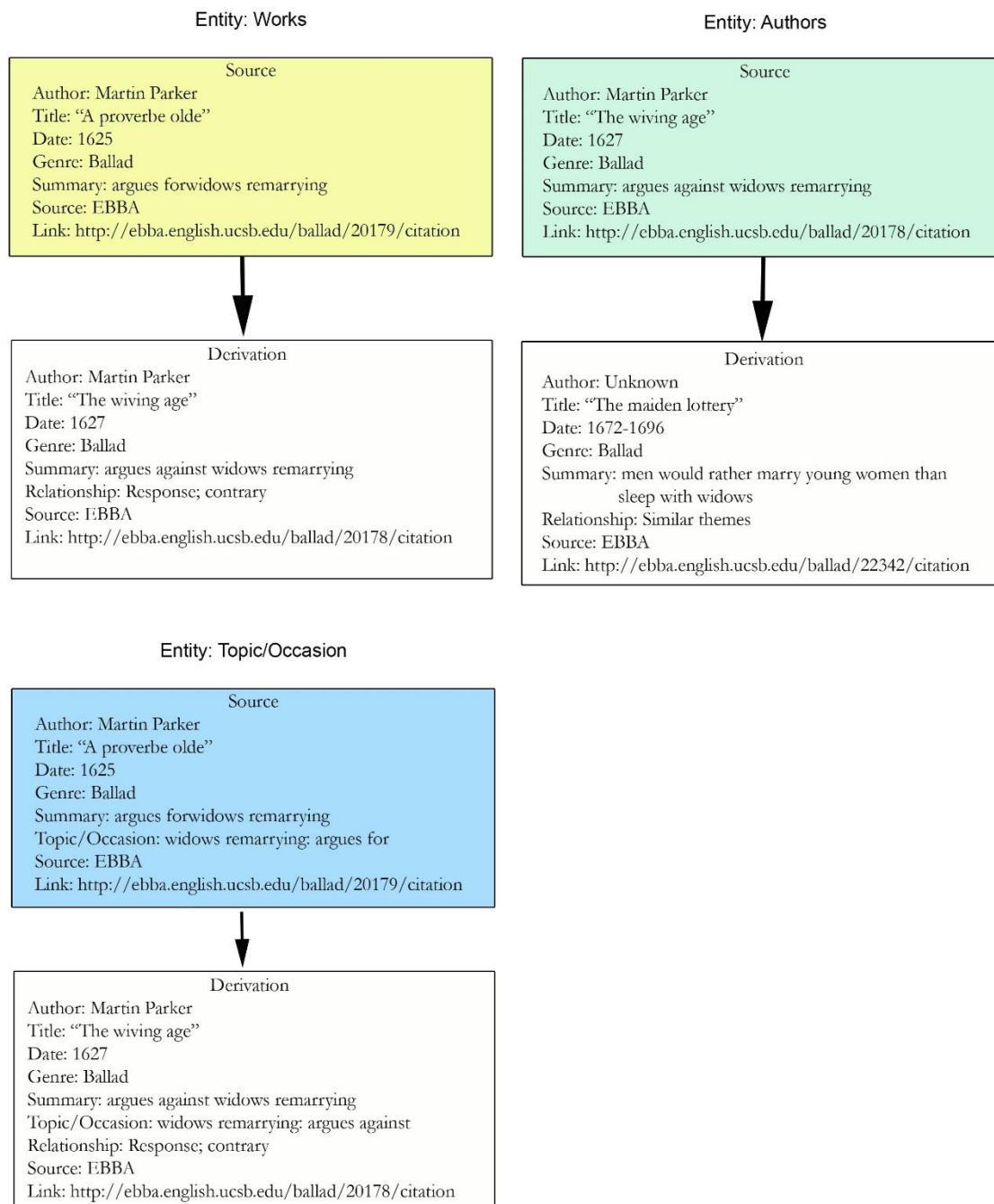


Figure 3. Example of a network of derivative works based on the English Broadside Ballad Archive (EBBA).

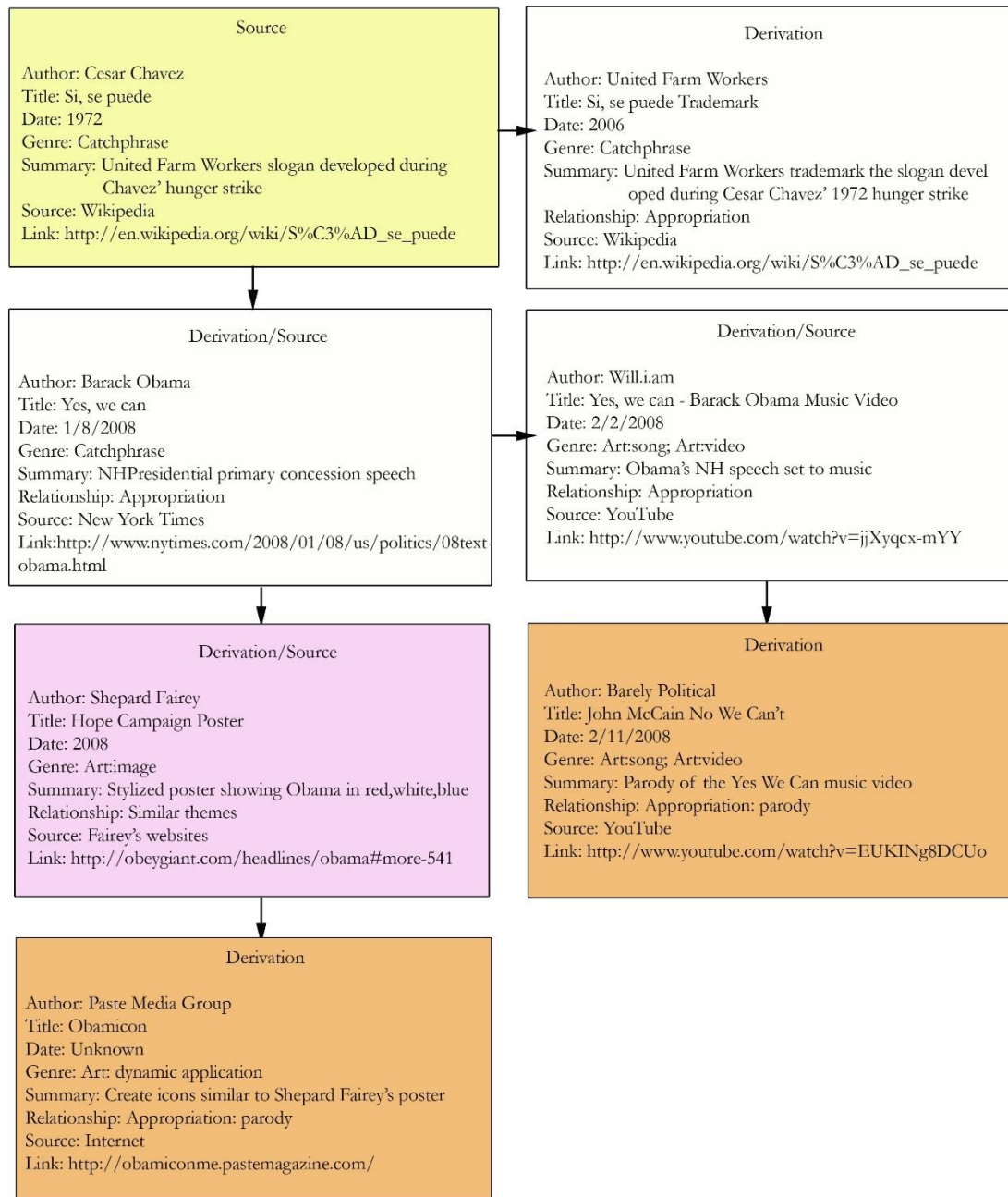


Figure 4. Example of a heterogeneous information network of derivative works based on the meme *Sí, se puede* / *Yes, We Can*.

Figure 5 depicts the data flow and workflow model for our system for *Extracting the Network of Influence in the Digital Humanities (ENIDH)*, as a block diagram. The system described in this book chapter implements a simplified variant of this workflow. On the left side, the input consists of candidate items to be compared –in this case, digital documents such as song videos bearing metadata. Named entity (NE) recognition and discovery plus terminology discovery are preliminary steps to relation discovery. As described in Section 5.2, supervised learning to predict parody/original song pairs

was conducted using a variety of inducers, but not using support vector machines (SVM) and other kernel-based methods. The desired web of influence (Koller, 2001) is represented by a heterogeneous information network (containing multiple types of entities such as “original song” and “parody video” or “original video” and “parody lyrics”) as illustrated in Figure 3 and Figure 4.

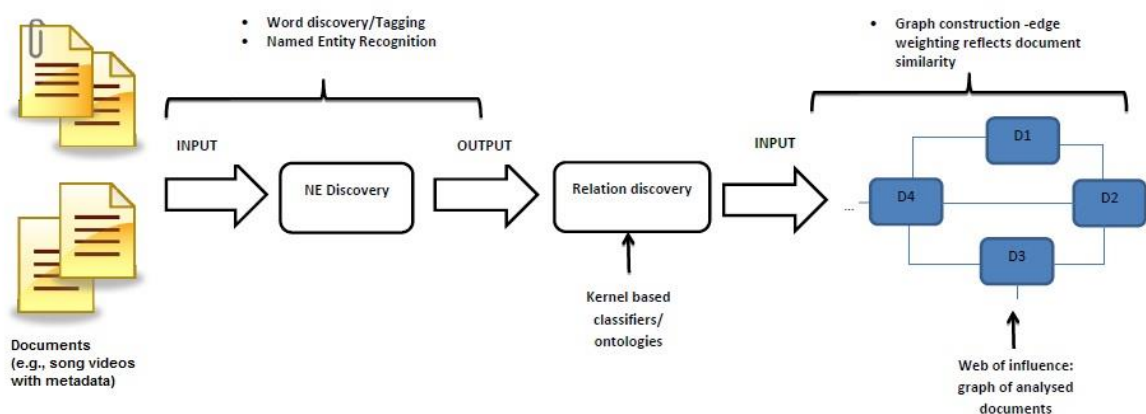


Figure 5. System block diagram: Extracting the Network of Influence in the Digital Humanities (ENIDH).

7. Acknowledgements

We thank the anonymous reviewers for helpful comments, and Hui Wang and Niall Rooney for the survey of kernel methods for clustering and classification of text documents in Section 2.3.

References

- (2014). Retrieved from Kimono Labs: <https://www.kimonolabs.com/>
- Alpaydin, E. (2014). *Introduction to Machine Learning, 3rd Edition*. Cambridge, MA, USA: MIT Press.
- API Overview Guide*. (2014, 4 6). Retrieved from Google Developers: <https://developers.google.com/youtube/>
- Blei, D. M., & Ng, A. Y. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 2003(3), 993-1022.
- Bloehdorn, S., & Moschitti, A. (2007). Combined syntactic and semantic kernels for text classification. *Advances in Information Retrieval*, 307-318.

- Bontcheva, K., Derczynski, L., Funk, A., Greenwood, M. A., Maynard, D., & Aswani, N. (2013). TwitIE: An Open-Source Information Extraction Pipeline for Microblog Text. *Proceedings of the International Conference on Recent Advances in Natural Language Processing*.
- Bull, S. (2010). *Automatic Parody Detection in Sentiment Analysis*.
- Bunescu, R., & Mooney, R. (2005). A shortest path dependency kernel for relation extraction. *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 724-731.
- Burfoot, C., & Baldwin, T. (2009). Automatic Satire Detection : Are You Having a Laugh ? *ACL-IJCNLP*, (pp. 161-164). Suntec, Singapore.
- Cadez, I., Heckerman, D., Meek, C., Smyth, P., & White, S. (2000). Visualization of navigation patterns on a Web site using model-based clustering. In R. Ramakrishnan, S. J. Stolfo, R. J. Bayardo, & I. Parsa (Ed.), *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2000)*, (pp. 280-284). Boston, MA, USA.
- Cancedda, N., Gaussier, E., Goutte, C., & Renders, J. (2003). Word sequence kernels. *The Journal of Machine Learning Research*, 3, 1059-1082.
- Caragea, D., Bahrwani, V., Aljandal, W., & Hsu, W. H. (2009). Ontology-based link prediction in the LiveJournal social network. In V. Bulitko, & J. C. Beck (Ed.), *Proceedings of the 8th Symposium on Abstraction, Reformulation and Approximation (SARA 2009)*. Lake Arrowhead, CA, USA.
- Choudury, S., & Breslin, J. G. (2010). *User Sentiment Detection: A YouTube Use Case*.
- Collins, M., & Duffy, N. (2002). Convolution kernels for natural language. *Advances in Neural Information Processing Systems*, 1, 625-632.
- Culotta, A., & Sorensen, J. (2004). Dependency tree kernels for relation extraction. *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 423-es.
- Cuzzocrea, A., Song, I.-Y., & Davis, K. C. (2011). Analytics over large-scale multidimensional data: the big data revolution! In A. Cuzzocrea, I.-Y. Song, & K. C. Davis (Ed.), *Proceedings of the ACM 14th International Workshop on Data Warehousing and On-Line Analytical Processing (DOLAP 2011)* (pp. 101-104). Glasgow, UK: ACM Press.
- Dawkins, R. (2000). Foreword. In S. Blackmore, *The Meme Machine* (pp. i-xvii). Oxford: Oxford University Press.
- Dawkins, R. (2006). *The Selfish Gene* (30th anniversary edition ed.). Oxford: Oxford University Press.
- Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., & Weischedel, R. (2004). The automatic content extraction (ace) program—tasks, data, and evaluation. *Proceedings of LREC*, 4, 837-840.

- Drummond, C., & Holte, R. E. (2012, 4 6). *Severe Class Imbalance: Why Better Algorithms Aren't the Answer*. Retrieved from <http://www.csi.uottawa.ca/~cdrummon/pubs/ECML05.pdf>
- Elger, C. E., & Lehnertz, K. (1998, February). Seizure prediction by non-linear time series analysis of brain electrical activity. *European Journal of Neuroscience*, 10(2), 786–789.
- Elshamy, W., & Hsu, W. H. (2014). Continuous-Time Infinite Dynamic Topic Models: The Dim Sum Process for Simultaneous Topic Enumeration and Formation. In W. H. Hsu, *Emerging Methods in Predictive Analytics: Risk Management and Decision-Making* (pp. 187-222). Hershey, PA, USA: IGI Global.
- Gargi, U., Lu, W., Mirrokni, V., & Yoon, S. (2011). Large-Scale Community Detection on YouTube for Topic Discovery and Exploration. In L. A. Adamic, R. A. Baeza-Yates, & S. Counts (Ed.), *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*.
- Gill, P., Arlitt, M., Li, Z., & Mahanti, A. (2007). YouTube Traffic Characterization: A View From the Edge. *IMC'07: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement* (pp. 15-28). New York, NY, USA: ACM.
- Gleick, J. (2011). *The Information: A History, a Theory, a Flood*. New York: Pantheon Books.
- Goldstein, J., & Roth, S. F. (1994). Using aggregation and dynamic queries for exploring large data sets. In E. Dykstra-Erickson, & M. Tscheligi (Ed.), *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2004)* (pp. 23-29). Boston, MA, USA: ACM Press.
- Google. (2012, 5 1). *Statistics*. Retrieved from YouTube: http://www.youtube.com/t/press_statistics
- Hall, M., Frank, E., Holmes, G., & Pfahringer, B. (2009, June). The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1), pp. 10-18.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1).
- Heer, J., Kong, N., & Agrawala, M. (2009). Sizing the horizon: the effects of chart size and layering on the graphical perception of time series visualizations. *Proceedings of the 27th International Conference on Human Factors in Computing Systems (CHI 2009)* (pp. 1303-1312). Boston, MA, USA: ACM Press.
- Hovy, E., & Lavid, J. (2010). Towards a 'Science' of Corpus Annotation: A New Methodological Challenge for Corpus Linguistics. *International Journal of Translation*, 22(1), 13-36. doi:10.1075/target.22.1
- Hsu, W. H., Lancaster, J. P., Paradesi, M. S., & Weninger, T. (2007). Structural Link Analysis from User Profiles and Friends Networks: A Feature Construction Approach. In N. S. Glance, N. Nicolov, E. Adar, M. Hurst, M. Liberman, & F. Salvetti (Ed.), *Proceedings of the 1st International Conference on Weblogs and Social Media (ICWSM2007)*, (pp. 75-80). Boulder, CO, USA.

- Jenkins, H. (2009, 02 27). *If it doesn't spread, it's dead*. Retrieved 06 16, 2011, from Confessions of an Aca-Fan: The Official Weblog of Henry Jenkins:
http://www.henryjenkins.org/2009/02/if_it_doesnt_spread_its_dead_p.html
- Keim, D. A. (2006). Challenges in Visual Data Analysis. In E. Banissi, K. Börner, C. Chen, G. Clapworthy, C. Maple, A. Lobben, . . . J. Zhang (Ed.), *10th International Conference on Information Visualisation (IV 2006)* (pp. 9-16). London, UK: IEEE Press.
- Koller, D. (2001). *Representation, Reasoning, Learning: IJCAI 2001 Computers and Thought Award Lecture*. Retrieved from Daphne Koller: <http://stanford.io/TFV7qH>
- Krishna, A., Zambreno, J., & Krishnan, S. (2013). Polarity Trend Analysis of Public Sentiment on YouTube. *The 19th International Conference on Management of Data*. Ahmedabad, India.
- Kumar, N., Keogh, E., Lonardi, S., & Ratanamahatana, C. A. (2005). Time-series bitmaps: a practical visualization tool for working with large time series databases. *Proceedings of the 5th SIAM International Conference on Data Mining (SDM 2005)*, (pp. 531-535). Newport Beach, California, USA.
- Lieberman, M. (2003). *Penn Treebank POS*. Retrieved 2014, from Penn Arts and Sciences:
https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html
- LiteraryDevices Editor. (2014). Retrieved from Literary Devices: <http://literarydevices.net>
- Liu, J., Ali, S., & Shah, M. (2008). Recognizing Human Actions Using Multiple Features. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008)*, (pp. 1 - 8). doi:10.1109/CVPR.2008.4587527
- Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., & Watkins, C. (2002). Text classification using string kernels. *The Journal of Machine Learning Research*, 2, 419-444.
- Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 55-60.
- Mario, C., & Talia, D. (2003, January). The knowledge grid. *Communications of the Association for Computing Machinery*, 46(1), 89-93 .
- McCallum, A. K. (2002). Retrieved from MALLET: A Machine Learning for Language Toolkit:
<http://mallet.cs.umass.edu>
- Mesaros, A., & Virtanen, T. (2010). Automatic Recognition of Lyrics in Singing. *EURASIP Journal on Audio, Speech, and Music Processing*, 2010. doi:doi:10.1155/2010/546047
- Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the*

4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2, 2, 1003-1011.

Mitchell, T. M. (1997). *Machine Learning*. New York, NY, USA: McGraw Hill.

Monmonier, M. (1990). Strategies For The Visualization Of Geographic Time -Series Data. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 27(1), 30-45.

Moschitti, A. (2004). A study on convolution kernels for shallow semantic parsing. *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 335-es.

Moschitti, A. (2006). Syntactic kernels for natural language learning: The semantic role labeling case. *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers on XX*, (pp. 97-100).

Moschitti, A. (2008). Kernel methods, syntax and semantics for relational text categorization. *Proceeding of the 17th ACM Conference on Information and Knowledge Management*, (pp. 253-262).

Moschitti, A., Pighin, D., & Basili, R. (2008). Tree kernels for semantic role labeling. *Computational Linguistics*, 34(2), 193-224.

Murphy, J. C., Hsu, W. H., Elshamy, W., Kallumadi, S., & Volkova, S. (2014). Greensickness and HPV: A Comparative Analysis? In T. Gniady, K. McAbee, & J. C. Murphy, *New Technologies in Renaissance Studies II* (Vol. 4, pp. 171-197). Toronto and Tempe, AZ, USA: Iter and Arizona Center for Medieval and Renaissance Studies.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press.

Nguyen, T., Moschitti, A., & Riccardi, G. (2009). Convolution kernels on constituent, dependency and sequential structures for relation extraction. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3, 3*, 1378-1387.

O'reilly, T. (2009). *What is web 2.0*. O'Reilly Media, Inc.

Reyes, A., Rosso, P., & Veale, T. (2012). A Multidimensional Approach for Detecting Irony in Twitter. *Language Resources and Evaluation*, 47(1), 239-268.

Selden, J. (1689). *Table Talk: Being the Discourses of John Selden*. London: Printed for E. Smith.

Shawe-Taylor, J., & Cristianini, N. (2004). *An introduction to support vector machines: And other kernel-based learning methods*. Cambridge, UK: Cambridge University Press.

Siersdorfer, S. C. (2010). How Useful are Your Comments ? - Analyzing and Predicting YouTube Comments and Comment Ratings. *Proceedings of the 19th International Conference on World Wide Web*, 15, pp. 897-900.

- Simmonet, V. (2013). Classifying YouTube Channels: a practical system. *Proceedings of the 22nd International Conference on World Wide Web Companion*, (pp. 1295-1303).
- Steele, J., & Iliinsky, N. (Eds.). (2010). *Beautiful Visualization: Looking at Data through the Eyes of Experts*. Cambridge, MA, USA: O'Reilly Media.
- Trindade, L. A., Wang, H., Blackburn, W., & Rooney, N. (2011). Text classification using word sequence kernel methods. *Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC 2011)*, (pp. 1532-1537). Guilin, China.
- Trindade, L. A., Wang, H., Blackburn, W., & Taylor, P. S. (2014). Enhanced Factored Sequence Kernel for Sentiment Classification. *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies (WI-IAT 2014)*, (pp. 519-525).
- Tsur, O., Davidov, D., & Rappoport, A. (2010). *ICWSN - A Great Catchy Name: Semi-supervised Recognition of Sarcastic Sentences in Online Product Reviews*. AAAI.
- Wang, M. (2008). A re-examination of dependency path kernels for relation extraction. *Proceedings of IJCNLP*, 8.
- Watson, H. J., & Wixom, B. H. (2007, September). The Current State of Business Intelligence. *IEEE Computer*, 40(9), 96-99.
- Watt, T. (1991). *Cheap Print and Popular Piety, 1550-1640*. Cambridge, UK: Cambridge University Press.
- Wattenhofer, M., Wattenhofer, R., & Zhu, Z. (2012). The YouTube Social Network. *Sixth International AAAI Conference on Weblogs and Social Media*, (pp. 354-361).
- Weese, J. L. (2014). Predictive Analytics in Digital Signal Processing: A Convolutional Model for Polyphonic Instrument Identification and Pitch Detection Using Combined Classification. In W. H. Hsu, *Emerging Methods in Predictive Analytics: Risk Management and Decision-Making* (pp. 223-253). Hershey, PA, USA: IGI Global.
- Yang, M., Hsu, W. H., & Kallumadi, S. (2014). Predictive Analytics of Social Networks: A Survey of Tasks and Techniques. In W. H. Hsu, *Emerging Methods in Predictive Analytics: Risk Management and Decision-Making* (pp. 297-333). Hershey, PA, USA: IGI Global.
- Yang, W., & Toderici, G. (2011). Discriminative Tag Learning on YouTube Videos with Latent Sub-tags. *CVPR*, 3217-3224.
- Yoganarasimhan, H. (2012, 4 6). *Impact of Social Network Structure on Content Propagation: A Study Using YouTube Data*. Retrieved from <http://faculty.gsm.ucdavis.edu/~hema/youtube.pdf>
- Zelenko, D., Aone, C., & Richardella, A. (2003). Kernel methods for relation extraction. *The Journal of Machine Learning Research*, 3, 1083-1106.

Zhang, M., Zhang, J., Su, J., & Zhou, G. (2006). A composite kernel to extract relations between entities with both flat and structured features. *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, 825-832.