# YouCat : Weakly Supervised Youtube Video Categorization System from Meta Data & User Comments using WordNet & Wikipedia

*Subhabrata Mukherjee*[†], *Pushpak Bhattacharyya*[‡]
[†]IBM India Research Lab
[‡]Dept. of Computer Science and Engineering, IIT Bombay
subhabmu@in.ibm.com, pb@cse.iitb.ac.in

ABSTRACT

In this paper, we propose a *weakly supervised* system, *YouCat*, for categorizing Youtube videos into different genres like *Comedy, Horror, Romance, Sports* and *Technology* The system takes a *Youtube video url* as input and gives it a belongingness score for each genre. The key aspects of this work can be summarized as: (1) Unlike other genre identification works, which are *mostly* supervised, this system is *mostly unsupervised*, requiring *no labeled data for training*. (2) The system can easily incorporate new genres without requiring labeled data for the genres. (3) YouCat extracts information from the *video title, meta description* and *user comments* (which together form the *video descriptor*). (4) It uses *Wikipedia* and *WordNet* for concept expansion. (5) The proposed algorithm with a time complexity of *O(|W|)* (where (|W|) is the number of words in the video descriptor) is efficient to be deployed in web for real-time video categorization. Experimentations have been performed on real world Youtube videos where YouCat achieves an F-score of *80.9%*, without using any labeled training set, compared to the supervised, multiclass SVM F-score of *84.36%* for *single genre prediction*. YouCat performs better for multi-genre prediction with an F-Score of *90.48%*. Weak supervision in the system arises out of the usage of manually constructed WordNet and genre description by *a few root words.*

KEYWORDS : Youtube, Genre Prediction, Comments, Metadata, Wikipedia, WordNet

# 1 INTRODUCTION

In recent times there has been an explosion in the number of online videos. With the gradually increasing multimedia content, the task of efficient query-based video retrieval has become important. The proper genre or category identification of the video is essential for this purpose. The automatic genre identification of videos has been traditionally posed as a supervised classification task of the features derived from the audio, visual content and textual features. Whereas some works focus on classifying the video based on the *meta data (text)* provided by the uploader (Cui *et al.*, 2010; Borth *et al.*, 2009, Filippova *et al.*, 2011), other works attempt to extract low-level features by analyzing the *frames, signals, audio etc.* along with textual features (Ekenel *et al.*, 2010; Yang *et al.*, 2007). There have been some recent advances in incorporating new features for classification like the social content comprising of the *user connectivity* (Zhang *et al.*, 2011; Yew *et al.*, 2011), *comments* (Filippova *et al.*, 2011), *interest etc.*

All the above approaches pose the genre prediction task as supervised classification requiring a large amount of training data. It has been argued that a *serious challenge* for supervised classification is the availability and requirement of manually labeled data (Filippova *et. al*, 2010; Wu *et. al*, 2010; Zanetti *et. al*, 2008). For example, consider a video with the descriptor "*It's the NBA's All-Mask Team!*". Unless there is a video in the training set with *NBA* in the video descriptor labeled with *Sport*, there is no way of associating *NBA* to *Sport*. It is also not possible to associate *NBA* to *Basketball* and then to *Sport*.  As *new genre-related concepts* (like new sports, technologies, domain-dependent terms *etc.*) appear every day the training set should expand incorporating all these new concepts, which makes training very expensive. As the number of categories or genres is increased the data requirement goes up compounded. The problem is enhanced by the *noisy* and *ambiguous* text prevalent in the media due to the *slangs, acronyms etc*. The *very short text* provided by the user, for *title* and *video description*, provide little context for classification (Wu *et al.*, 2012). The focus of this paper is to propose a system that requires no labeled data for training and can be easily extended to identify new categories. The system can easily adapt to changing times, incorporating world knowledge, to overcome the labeled data shortage. It extracts all the features from the video uploader provided meta-data like the *video title*, *description of the video* as well as the *user comments*. The system incorporates social content by analyzing the user comments on the video, which is essential as the meta-data associated with a video is often absent or not adequate enough to predict its category. *WordNet* and *Wikipedia* are used as world knowledge sources for *expanding* the video descriptor since the uploader provided text is frequently very short, as are the user comments. WordNet is used for knowing the meaning of an unknown word whereas Wikipedia is used for recognizing the *named entities* (which are mostly absent in the WordNet) like "*NBA*" in the given example. In this work, we show how the textual features can be analyzed with the help of WordNet and Wikipedia to predict the video category without requiring any labeled training set.

The only weak supervision in the system arises out of the usage of a root words list (~ 1-3 words) used to describe the genre, WordNet which is manually annotated and a simple setting of the parameters of the model.

The paper is organized as follows: Section 2 gives the related work and compares them with our approach. Section 3 discusses the unsupervised feature extraction from various sources. Section 4 gives the algorithm for feature vector classification and genre identification. Section 5 discusses the parameter settings for the model. The experimental evaluations are presented in Section 6 followed by discussions of the results in Section 7. Section 8 concludes the paper with future works and conclusions.

## 2 RELATED WORK

The video categorization works can be broadly divided under 3 umbrellas: 1. Works that deal with low level features by extracting features from the video frames like the audio, video signals, colors, textures *etc.* 2. Works that deal with textual features like the title, tag, video description, user comments *etc.* 3. Works that combine low-level features like the video frame information with the high-level text features. In this section, we discuss only those works that include text as one of the features. Our work is similar to text classification but for a different application.

Filippova *et al.* (2011) showed that a text-based classifier, trained on imperfect predictions of weakly supervised video content-based classifiers, outperforms each of them taken independently. They use features from the video title, description, user comments, uploader assigned tags and use a maximum entropy model for classification.

Wang *et al.* (2010) considers features from the text as well as low-level video features, and proposes a fusion framework in which these data sources are combined with the small manually-labeled feature set independently. They use a Conditional Random Field (CRF) based fusion strategy and a Tree-DRF for classification.

The content features are extracted from training data in Cui *et al.* (2010) to enrich the text based semantic kernels to yield content-enriched semantic kernel which is used in the SVM classifier.

Borth *et al.* (2009) combines the results of different modalities like the uploader generated tags and visual features which are combined using a weighted sum fusion, where SVM's are used with bag of words as features. These categories are refined further by deep-level clustering using probabilistic latent semantic analysis.

Query expansion is performed in Wu *et al.* (2012) by using contextual information from the web like the related videos and user videos, in addition to the textual features and use SVM in the final phase for classification.

Some works have used user information like the browsing history along with other textual features. Zhang *et al.* (2006) develop a video categorization framework that combined multiple classifiers based on normal text features as well as users' querying history and clicking logs. They used Naïve Bayes with a mixture of multinomials, Maximum Entropy, and Support Vector Machines for video categorization.

Most of the works are similar to Huang *et al.* (2010) which use different text features and classifiers like the Naïve Bayes, Decision Trees and SVM's for classification.

Yang *et al*. (2007) propose a semantic modality that includes concept histogram, visual word vector model and visual word Latent Semantic Analysis (LSA); and a text modality that includes titles, descriptions and tags of web videos. They use various classifiers such as Support Vector Machine(SVM), Gaussian Mixture Model(GMM) and Manifold Ranking (MR) for classification.

Song *et al*. (2009) developed an effective semantic feature space to represent web videos consisting of concepts with small semantic gap and high distinguishing ability where Wikipedia is used to diffuse the concept correlations in this space. They use SVM's with fixed number of support vectors (n-ISVM) for classification.

All the above works build on supervised classification systems, requiring labeled data for training, mostly using the Support Vector Machines. In this paper, we propose a system that requires no labeled data for training, which is the primary difference of our work with those surveyed. Also, the usefulness of Wikipedia and WordNet for concept expansion has not been probed much in earlier video categorization tasks, save a few. We use many of the ideas from the above works and integrate them into YouCat.

# 3 FEATURE CONSTRUCTION

Given a *Youtube video url*, the objective is to assign scores to it which represent its belongingness to the different genres. The video genres are categories like *romance, comedy, horror, sports* and *technology*. The genre names are pre-defined in the system along with a small set of *root words* for each genre. The root words act like a description of the genre. For example, *funny* and *laugh* act as the key characteristics of the *comedy* genre. This allows *new* genres to be easily defined in the system in terms of the root words as well as to have a fine distinction between the genres.

A *seed list* of words is *automatically* created for each genre by searching a *thesaurus* using the roots words for that genre. A *concept list* is created for each genre with *relevant words* from the *WordNet* and *named entities* in *Wikipedia,* with the help of the seed list of the corresponding genre. Given a video descriptor consisting of the *video title*, the *meta-description of the video* and the *user comments*, the seed list and the concept list for each genre are used for finding appropriate matches in the *video descriptor* to predict appropriate tags or categories for the video using the scores.

## 3.1 Data Pre-Processing

### 3.1.1 Seed List Creation using Root Words

A set of tags is pre-defined in the system along with a set of 1-3 *root words* for each tag. A *seed list* of words is created for each genre (*defined in the system*) which captures the key characteristics of that category. For Example, *"love", "hug", "cuddle" etc.* are the characteristics of the *Romance* genre. *Root words of the genre* are taken and all their synonyms are retrieved from a thesaurus. *The root words list and the genre names are pre-defined in the system. Table 1* shows the root-words list for the *five* genres used in this work. An automatic breadth-first search is done on the thesaurus based on the root words to retrieve only the most relevant synonyms or associated concepts. For example, the word *Laugh* is taken for its genre *Comedy* and all its first level synonyms are retrieved which are again recursively used to retrieve their level-one synonyms till a certain depth. A thesaurus is used for this purpose which gives every day words and slangs. In our work, the following thesaurus[1] retrieves the words *rofl, roflmao, lol etc.* when the word *Laugh* is looked up from the *Comedy* genre. A *snapshot* of the seed lists with number of words in the lists is shown in *Table 2*.

The set of root words can help in *fine genre distinction* as the seed list will have only *associated* concepts. For example if the *Transport* genre is sub-categorized into *Road* and *Railways*, the corresponding root words {car, road, highway, auto} and {train, rail, overhead wire, electricity, station} will distinguish between the two.

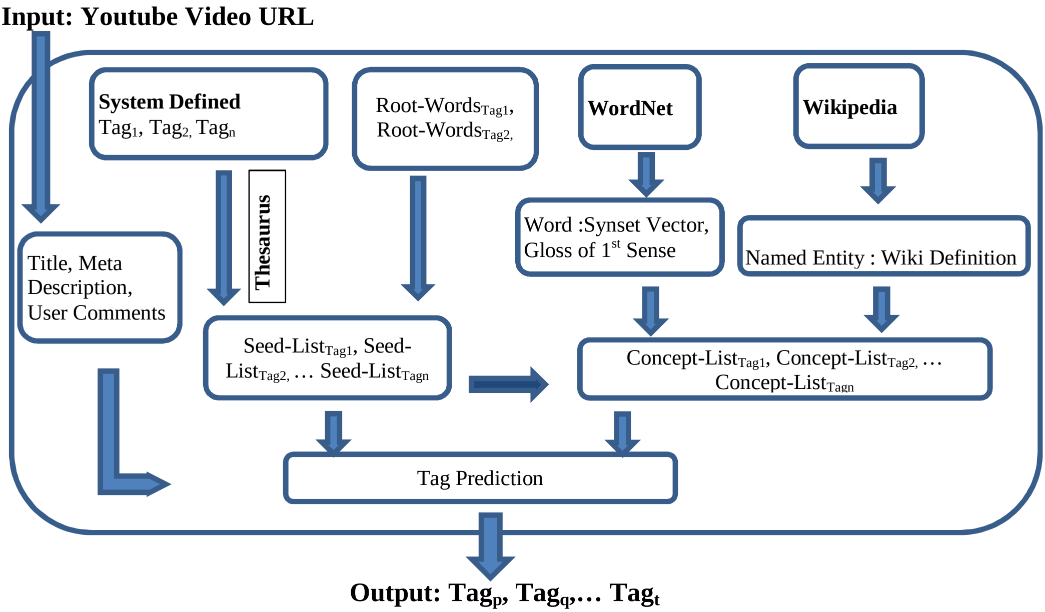

**Input: Youtube Video URL**
**System Defined**
Tag$_1$, Tag$_2$, Tag$_n$
Root-Words$_{Tag1}$,
Root-Words$_{Tag2,}$
**WordNet**
**Wikipedia**
**Thesaurus**
Word :Synset Vector,
Gloss of 1$^{st}$ Sense
Named Entity : Wiki Definition
Title, Meta
Description,
User Comments
Seed-List$_{Tag1}$, Seed-List$_{Tag2,}$ … Seed-List$_{Tagn}$
Concept-List$_{Tag1}$, Concept-List$_{Tag2,}$ …
Concept-List$_{Tagn}$
Tag Prediction
**Output: Tag$_p$, Tag$_q$,… Tag$_t$**

**Fig. 1.** System Block Diagram

| | |
|---|---|
| Comedy | comedy, funny, laugh |
| Horror | horror, fear, scary |
| Romance | romance, romantic |
| Sport | sport, sports |
| Technology | tech, technology, science |

**Table 1.** Root Words for Each Genre

[1] www.urbandictionary.com/thesaurus.php

| | |
|---|---|
| Comedy (25) | funny, humor, hilarious, joke, comedy, roflmao, laugh, lol, rofl, roflmao, joke, giggle, haha, prank |
| Horror (37) | horror, curse, ghost, scary, zombie, terror, fear, shock, evil, devil, creepy, monster, hell, blood, dead, demon |
| Romance (21) | love, romantic, dating, kiss, relationships, heart, hug, sex, cuddle, snug, smooch, crush, making out |
| Sports (35) | football, game, soccer, basketball, cheerleading, sports, baseball, FIFA, swimming, chess, cricket, shot |
| Tech (42) | internet, computers, apple, iPhone, phone, pc, laptop, mac, iPad, online, google, mac, laptop, XBOX, Yahoo |

**Table 2.** Snapshot of Seed List for Each Genre

### 3.1.2 Concept Hashing

Each word (used as a *key* for hashing) in the WordNet, that is not present in any seed list, is hashed with the set of all its synsets and the gloss of its first sense.

A synset is a set of synonyms that collectively disambiguate each other and give a unique sense to the set. For example, the word *dunk* has the synsets - {*dunk, dunk* <u>*shot*</u>*, stuff* <u>*shot*</u>**;** *dunk, dip, souse, plunge, douse***;** *dunk***;** *dunk, dip*}. Here the first synset {*dunk, dunk shot, stuff shot*} has the sense of a basketball shot. The meaning of a synset is clearer with its gloss. A gloss[2] is the definition or example sentences for a synset which portrays the context in which the synset or sense of the word can be used. For example, the gloss of the synset {*dunk, dunk shot, stuff shot*} is {*a basketball shot in which the basketball is propelled downward into the basket*}.

Technically, we should have taken only the words in the synset of its most appropriate sense. But we do not perform word sense disambiguation[3] to find out the proper synset of the word. Taking only the first sense provides fewer contexts while classifying the feature vector, and so the information from all the senses of a given word is used. The gloss of the first sense is frequently used, as in many cases the first sense is the best sense of a word (Macdonald *et al.*, 2007).

Wikipedia is necessary for *named entity recognition*, since the WordNet does not contain most of these entries. All the *named entities* in Wikipedia with the *top 2 line definition* in their corresponding Wiki articles are stored in a hashtable. For example, *NBA* is stored in the hashtable with its definition from the Wikipedia article as {*The National* <u>*Basketball*</u> *Association (NBA) is the pre-eminent men's professional* <u>*basketball*</u> *league in North America. It consists of thirty franchised member clubs, of which twenty-nine are located in the United States and one in Canada.*}.

Most of the named entities in practice are not unigrams like *Michael Jordon*. If the unigrams in this named entity are expanded separately, a different sense for each would be retrieved. This is not desirable. In this work, we use a simple heuristics method based on *capitalization* of the

[2] http://en.wikipedia.org/wiki/WordNet
[3] http://en.wikipedia.org/wiki/Word-sense_disambiguation

letters to identify the named entities. Any sequence of consecutive words such that each of them starts with a capital letter, and the sequence does not start or end with any *Stop Word* is considered a named entity. Stop Words are allowed within this sequence, provided the number of such Stop Words between any two consecutive words is less than or equal to two. Thus named entities like *United States of America, Lord of the Rings, Bay of Bengal etc.* are recognized. This method captures a lot of false positives. One such example can be the usage of capitalization in the social media in the form of pragmatics to express the intensity of emotions (Example: *I just LOVED that movie*). However, false positives are not a concern in our case as such entries, *if valid*, will only add to the concept lists. The named entity is considered as a single token and treated just like the unigrams.

## 3.2 Concept List Creation

Let *w* be any given word and its expanded form given by WordNet (*set of all its synsets* and the *gloss of its first sense*) or Wikipedia (*top 2 line definition*) be denoted by $w'$. Let $w'_j$ be the j$^{\text{th}}$ word in the expanded word vector. Let $seed_k$ and $root_k$ be the seed list and root words list, respectively, corresponding to the $k^{th}$ genre. The genre of *w* is given by

$$genre(w) = argmax_k \sum_j \mathbf{1}_{w'_j \in seed_k,\, w'_j \in root_k}$$

$$\ldots Equation\ 1$$

Here, **1** is an indicator function which returns 1 if a particular word is present in the seed list or root words list corresponding to a specific genre and 0 otherwise. In the given example, with the pre-defined 5 genres (*Table 1*), *dunk* and *basketball* both will be classified to the *Sports* genre as they have the maximum matches (*"shot", "basketball"*) from the seed list corresponding to the *Sports* genre in their expanded concept vector.

Finally, a *concept list* is created for each genre containing *associated* words in the WordNet (ignoring those in the seed lists) and named entities in the Wikipedia.

## 3.3 Video Descriptor Extraction

Given a video url, the *video title*, the *meta description* of the video and all the *user comments* on the video from Youtube are retrieved. A *stopwords* list is used to remove words like *is, are, been etc.* A lemmatizer is used to reduce each word to its base form or lemma. Thus "*play*", "*played*", "*plays*", "*playing*" are reduced to its lemma "*play*".

Consider the sentence in a video descriptor, "*It was an awesome slam dunk in the NBA finals by Michael Jordan*". None of the words here is present in any seed list. But *dunk* and *NBA* are present in the concept list corresponding to *Sports* genre and thus the given sentence is associated to *Sports*. The association (*Sports* via *Basketball*) can also be captured by considering the named entity *Michael Jordon* in Wikipedia.

# 4 FEATURE VECTOR CLASSIFICATION

Let the video descriptor *f* consist of *n* words, in which the *j*$^{th}$ word is denoted by *word*$_j$. The *root word list, seed list* and the *concept list* for the $k^{th}$ genre are denoted by $root_k$, $seed_k$ and $concept_k$ respectively. The score of $f$ belonging to a particular $genre_k$ is given by,

$$score(f \in genre_k; w_1, w_2, w_3) = w_1 \times \textstyle\sum_j \mathbf{1}_{word_j \in root_k} + w_2 \times \sum_j \mathbf{1}_{word_j \in seed_k} + w_3 \times \sum_j \mathbf{1}_{word_j \in concept_k}$$

$$where \quad w_3 < w_2 < w_1 \qquad \ldots Equation\ 2$$

Here, **1** is an indicator function that returns 1 if a word is present in the root words list, seed list or concept list corresponding to $genre_k$ and 0 otherwise. Weights $w_1, w_2\ and\ w_3$ are assigned to words present in the root words list, seed list and the concept list respectively. The weight assigned to any root word is maximum as it has been specified as a part of the genre description manually. Lesser weightage is given to the words in the seed list as they are automatically extracted using a thesaurus. The weight assigned to concept list is the least to reduce the effect of *topic drift* during concept expansion (Manning *et al.*, 2008). The topic drift occurs due to the enlarged context window, during concept expansion, which may result in a match from the seed list of some other genre than the one it actually belongs to.

The score of a video belonging to a particular genre is,

$$score(video \in genre_k; p_1, p_2, p_3) = p_1 \times score(f^{Title} \in genre_k) + p_2 \times score(f^{Meta\ Data} \in genre_k) + p_3 \times score(f^{Comments} \in genre_k)$$

$$\ldots Equation\ 3$$

Here $p_1$, $p_2$, $p_3$ denote the weight of the feature belonging to the *title, meta data (meta description of the video)* and *user comments* respectively where $p_1 > p_2 > p_3$. This is to assign more importance to the title, then to the meta data and finally to the user comments. The genre to which the video belongs is given by,

$$video_{genre} = argmax_k\ score(video \in genre_k)$$

$$\ldots Equation\ 4$$

This assigns the highest scoring genre as the desired category for the video. However, most of the popular videos in Youtube can be attributed to more than one *genre*. Thus to allow multiple tags to be assigned to a video, a thesholding is done and the prediction is modified as:

$$video_{genre} = k, if\ score(video \in genre_k) \geq \theta$$
$$where\ \theta = \frac{1}{k} \sum_k score(video \in genre_k)$$

$$\ldots Equation\ 5$$

If the score of the video for any genre is greater than the average score of all the genres, then it is assigned as a possible tag for the video. In case the genre scores for the 5 categories are something like {*400, 200, 100, 50, 10*} with *avg=152*, then the first 2 genres are chosen. If any of the genre score is very high compared to the others, the average will rise decreasing the chance of other genres being chosen. *Algorithm 1* describes the genre identification steps in short.

```
Pre-processing:
    1. Define Genres and Root Words List for each genre
    2. Create a Seed list for each genre by breadth-first-search in a Thesaurus,
       using root words in the genre or the genre name
    3. Create a Concept List for each genre using all the words in WordNet (not
       present in Seed Lists) and Named Entities in Wikipedia using Equation 1
Input: Youtube Video Url
    1. Extract Title, Meta Description of the video and User Comments from
       Youtube to form the video descriptor
    2. Lemmatize all the words in the descriptor removing stop word.
    3. Use Equations 2-4 for genre identification of the given video
Output: Genre Tags
```

**Algorithm 1.** Genre Identification of a Youtube Video

## 5 PARAMETER SETTING

The upweighting of document zones by giving more weightage to some portions of the text than others is common in automatic text summarization and information retrieval (Manning *et al.*, 2008). A common strategy is to use extra weight for words appearing in certain portions of the text like the title and use them as separate features, even if they are present in some other portion of the text (Giuliano *et al.*, 2011). As a rule-of-thumb the weights can be set as integral multiples, preferably prime, to reduce the possibility of ties (Manning *et al.*, 2008).

We follow this line of thought in our work and upweight certain portions of the text like the *title, meta data*, *user comments* separately. We also assign different weight to words belonging to different lists according to importance.

There are 6 parameters for the model we used: $w_1,\ w_2,\ w_3,\ p_1,\ p_2,\ p_3$. The parameters can be best trained if some label information is available. However, in the absence of any label information, we adopt a simple approach to parameter setting as mentioned above. We took the first set of integers, satisfying all the constraints in *Equations 2 and 3*, and assigned them to the 6 parameters: $w_1 = 3,\ w_2 = 2, w_3 = 1,\ p_1 = 3,\ p_2 = 2,\ p_3 = 1$.

**Semi-Supervised Learning of Parameters**

This work *does not* evaluate this dimension for parameter learning, since our objective has been to develop a system that requires no labeling information. However, if some category information is available, a robust learning of parameters is possible.

*Equation 1 and 2* can be re-written as:

$$score\left(f_k^{position} \in genre_k; w_1, w_2, w_3\right) = \ w_1 \times X_{1,k}^{position} + w_2 \times X_{2,k}^{position} + w_3 \times X_{3,k}^{position}$$

$$score(video_k \in genre_k; p_1, p_2, p_3) = Y_k = \sum_{position} p_{position} \sum_j w_j \times X_{j,k}^{position}$$

$= \sum_i \sum_j w'_{ij} X_j^i \quad (where\ w'_{ij} = p_i \times w_j)$

$$Or, Y_k = \boldsymbol{W}.\boldsymbol{X}_k\ (where\ \boldsymbol{W} = \left[w'_{1,1}\ w'_{1,2} \ldots w'_{3,3}\right]^T_{9\times1}, \qquad \boldsymbol{X}_k = [X^1_{1,k}\ X^1_{2,k} \ldots X^3_{3,k}]_{1\times9}$$

$$Or, \boldsymbol{Y} = \boldsymbol{W^T}.\boldsymbol{X}$$

This is a linear regression problem which can be solved by the ordinary least squares[4] method by minimizing the sum of the squared residuals i.e. the sum of the squares of the difference between the observed and the predicted values (Bishop *et al.*, 2006). The solution for $\boldsymbol{W}$ is given by $\boldsymbol{W} = (\boldsymbol{X^T X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$

A regularizer can be added to protect against over-fitting and the solution can be modified as: $\boldsymbol{W} = (\boldsymbol{X^T X} + \delta\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{Y} \quad where\ \delta\ is\ a\ parameter\ and\ \boldsymbol{I}\ is\ the\ identity\ matri$x.

# 6 EVALUATION

## 6.1 Data Collection

The following 5 genres are used for evaluation: *Comedy, Horror, Sports, Romance and Technology*. 12,837 videos are crawled from the Youtube following a similar approach like (Cu *et al.*, 2010; Wu *et al.*, 2012; Song *et al.*, 2009). Youtube has 15 pre-defined categories like *Romance, Music, Sports, People, Comedy etc*. These videos are automatically categorized in Youtube based on the user-provided tags while uploading the video and the video description. We crawl the videos directly from those categories using the Youtube API. *Table 3* shows the number of videos from each genre.

| Comedy | Horror | Sports | Romance | Tech | Total |
|---|---|---|---|---|---|
| 2682 | 2802 | 2577 | 2477 | 2299 | 12837 |

**Table 3:** Number of Videos in Each Genre

Only the 1st page of user comments is taken with comment length less than 150 characters. Short length comments are chosen as they are typically to the point, whereas long length comments often stray off the topic. The user comments are normalized by removing all the punctuations and reducing words like "*loveeee*" to "*love*". The number of user comments varied from 0 to 800 for different videos. *Table 4* shows the average number of user comments for the videos in each genre.

| Comedy | Horror | Sports | Romance | Tech |
|---|---|---|---|---|
| 226 | 186 | 118 | 233 | 245 |

**Table 4:** Average User Comments for Each Genre

The first integer values satisfying the constraints in the equations are taken as parameter values, which are set as: $w_1 = 3,\ w_2 = 2, w_3 = 1,\ p_1 = 3,\ p_2 = 2,\ p_3 = 1$.

[4] http://en.wikipedia.org/wiki/Ordinary_least_squares

## 6.2 Baseline System

All the words in the video descriptor consisting of the title, meta-description of the video and the user comments are taken as features for the SVM. A Multi-Class Support Vector Machines Classifier[5] with various features, like combination of unigrams and bigrams, incorporating part-of-speech (POS) information, removing stop words, using lemmatization *etc.*, is taken as the baseline. *Table 5* shows the baseline system accuracy with various features. A *linear kernel* is used with *10-fold* cross validation. SVM with lemmatized unigrams and bigrams as features, ignoring stop words, gave the maximum accuracy of *84.36%*.

| **SVM Features** | **$F_1$-Score(%)** |
|---|---|
| All Unigrams | 82.5116 |
| Unigrams+Without stop words | 83.5131 |
| Unigrams+ Without stop words +Lemmatization | 83.8131 |
| Unigrams+Without stop words +Lemmatization+ POS Tags | 83.8213 |
| Top Unigrams+Without stop words +Lemmatization+POS Tags | 84.0524 |
| All Bigrams | 74.2681 |
| **Unigrams+Bigrams+Without stop words+Lemmatization** | **84.3606** |

**Table 5:** Multi-Class SVM Baseline with Different Features

## 6.3 YouCat Evaluation

Experiments are performed on the videos *with* and *without user comments* as well as *with* and *without concept expansion*, to find out their effectiveness in video categorization. The system does not tag every video. It will not tag a video if it does not find a clue in the video descriptor that is present in the seed list or the concept list (*i.e.* the scores are all *zero*); or when there are ties with scores for multiple genres being equal. The precision, recall and $f_1$-score for each genre are defined as:

$$precision = \frac{number\ of\ videos\ correctly\ tagged}{number\ of\ videos\ tagged} \times 100$$

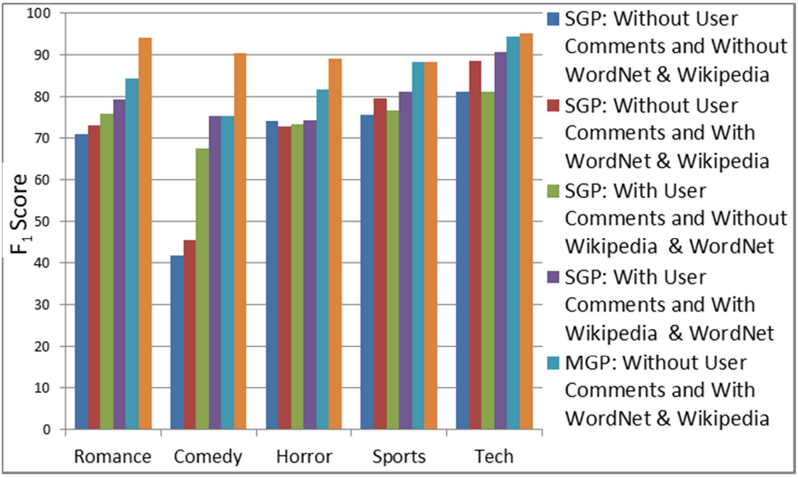
$$recall = \frac{number\ of\ video\ correctly\ tagged}{number\ of\ videos\ present\ in\ the\ genre} \times 100$$

$$f_1\ score = \frac{2 * precision * recall}{precision + recall}$$

*Graph 1* shows the incremental *$f_1$-score* improvement for each of the genres with and without *concept expansion* as well as with and without incorporating *user comments*. It also shows the genre-wise $f_1$-score improvement for multi-genre prediction model.

[5] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

The prediction is taken to be correct if the originally labeled tag is one of the predicted tags in a multi-genre prediction model. It may seem that the performance improvement for multiple genre identification, in our case, is trivial to achieve as the system can achieve 100% accuracy by simply assigning all the given genres to a video. This is because the prediction is taken to be correct if *any* of the predicted tags matches with the labeled tag. Thus an important performance measurement parameter is the *number of predicted tags* for each video. *Table 6* shows the average number of predicted tags for each video in each genre, with and without user comments.



SGP: Single Genre Prediction, MGP: Multiple Genre Prediction

Graph 1: Genre-wise $F_1$-Score Improvement for Different Models

| Genre | Average Tags/Video Without User Comments | Average Tags/Video With User Comments |
|---|---|---|
| Romance | 1.45 | 1.55 |
| Comedy | 1.67 | 1.80 |
| Horror | 1.38 | 1.87 |
| Sports | 1.36 | 1.40 |
| Tech | 1.29 | 1.40 |
| **Average** | **1.43** | **1.60** |

**Table 6:** Average Predicted Tags/Video in Each genre

*Table 7* shows the confusion matrix when single genre prediction is done with User Comments, Wikipedia & WordNet. *Table 8* shows average $f_1$-score for the different models used.

| **Genre** | **Romance** | **Comedy** | **Horror** | **Sports** | **Tech** |
|---|---|---|---|---|---|
| Romance | 80.16 | 8.91 | 3.23 | 4.45 | 3.64 |
| Comedy | 3.13 | 77.08 | 3.47 | 9.03 | 7.29 |
| Horror | 10.03 | 9.34 | 75.78 | 3.46 | 1.38 |
| Sports | 0.70 | 7.30 | 0 | 89.05 | 2.92 |
| Tech | 0.72 | 5.07 | 0.36 | 1.81 | 92.03 |

**Table 7:** Confusion matrix for Single Genre Prediction

| Model | Average $F_1$ Score |
|---|---|
| Multi-Class SVM Baseline: With User Comments | 84.3606 |
| Single Genre Prediction : Without User Comments + Without Wikipedia & WordNet | 68.76 |
| Single Genre Prediction : With User Comments + Without Wikipedia & WordNet | 74.95 |
| Single Genre Prediction : Without User Comments + With Wikipedia & WordNet | 71.984 |
| *Single Genre Prediction : With User Comments+ With Wikipedia &WordNet* | *80.9* |
| Multi Genre Prediction : Without User Comments + With Wikipedia & WordNet | 84.952 |
| **Multi Genre Prediction : With User Comments + With Wikipedia & WordNet** | **91.48** |

**Table 8:** Average $F_1$-Score of Different Models

# 7 EVALUATION

## 7.1 Multi-Class SVM Baseline

The SVM has been taken as the baseline as it is found to perform the best in text classification and video categorization works. Ignoring stop words in the feature vector improved the accuracy of SVM over the all-unigram feature space. Further accuracy improvement is achieved by lemmatization. This is because all the related unigram features like *laugh*, *laughed, laughing etc.* are considered as a single entry *laugh,* which reduces the sparsity of the feature space.

The part-of-speech information further increased accuracy, as they help in *crude* word sense disambiguation. Consider the word *haunt* which has a noun synset and gloss as {*haunt, hangout, resort, repair, stamping ground -- (a frequently visited place)*}. It also has 3 verb synsets where the first verb sense is {*haunt, stalk -- (follow stealthily or recur constantly and spontaneously to; "her ex-boyfriend stalked her"; "the ghost of her mother haunted her")*}. Using POS information, the word haunt will have two entries now corresponding to *Noun_haunt* and *Verb_haunt*. Although the second sense is related to the *Horror* genre, the first sense is not which can only be differentiated using the POS tags.

Top unigrams help in pruning the feature space and removing noise which helps in accuracy improvement. Using *only bigrams* however decreases the accuracy as many unrelated pairs are captured which do not capture the domain characteristics. Using bigrams along with unigrams gives the highest accuracy. This is because the entities like *Michael Jordon* can be used as features as a whole, unlike in unigrams.

## 7.2 Overall Accuracy

Our system could not beat the multi-class SVM baseline of *84.36%* in single genre prediction; but it nevertheless achieved an $f_1$ score of *80.9%, without using any labeled data for training.* The multiple genre prediction, however, beats the baseline with *91.48%* $f_1$ score.

## 7.3 Effect of User Comments

The user comments often introduce noise through the off-topic conversations, spams, abuses *etc.*; the slangs, abbreviations and pragmatics prevalent in the user posts make proper analysis difficult. However, an improvement of *6 percentage point* and *9 percentage point* in the $f_1$ score for single genre prediction (without and with concept expansion respectively) using the comments, suggest that the greater context provided by the user comments provide more clues about the genre to help in genre identification. The corresponding improvement in the multiple genre prediction using concept expansion is around *7 percentage point.*

When concept expansion is not used, user comments contribute a performance improvement of *5 percentage point* in Romance, *1 percentage point* in Sports and a huge *26 percentage point* in Comedy. This suggests that the user information mostly helps in identifying funny videos, as well as romantic videos to some extent. Horror videos undergo mild performance degradation by incorporating user comments. Using concept expansion, user comments contribute an accuracy improvement of *6 percentage point* in Romance, a huge *30 percentage point* in Comedy and *2 percentage point* in the other genres.

## 7.4 Effect of Concept Expansion

In the genre identification task, using a seed set for each genre runs the risk of *topic drift*. This may occur as a concept may be identified to belong to an incorrect genre due to off-topic words by considering a larger context. However, less weightage is given to concept expansion than to a direct match in the seed list to alleviate this risk. In single genre prediction using concept expansion, an $f_1$ score improvement of 3 *percentage point* (when user comments are not used) and 6 *percentage point* (when user comments are used) show that Wikipedia and WordNet help in identifying unknown concepts with the help of lexical and world knowledge.

When user comments are not used, concept expansion contributes a performance improvement of *3 percentage point* in Romance, *4 percentage point* in Comedy, Sports and 7 *percentage point* in Tech. This suggests that the external knowledge sources help in easy identification of new technological concepts. Horror videos undergo mild performance degradation. Using the comments, concept expansion contributes an improvement of *8 percentage point* in Comedy and

*9 percentage point* in Tech. Again, the performance improvement in Comedy using Wikipedia can be attributed to the identification of the concepts like *Rotfl, Lolz, Lmfao etc.*

## 7.5 Average Number of Tags per Video in Multiple Genre Prediction

The number of predicted tags in multiple genre identification for each video, on an average, is *1.43* and *1.6* in the two cases (without and with user comments). This suggests that mostly a single tag and in certain cases bi-tags are assigned to the video. It is also observed that the average number of tags per video increases when user comments are used. This is due to the greater contextual information available from user comments leading to genre overlap.

## 7.6 Confusion between Genres

The confusion matrix indicates that Romantic videos are frequently tagged as Comedy. This is often because many Romantic movies or videos have light-hearted Comedy in them, which is identifiable from the user comments. The Horror videos are frequently confused to be Comedy, as users frequently find them funny and not very scary. Both Sports and Tech videos are sometimes tagged as Comedy. The bias towards Comedy often arises out of the off-topic conversation between the users in the posts from the *jokes, teasing etc.* Overall, from the precision figures, it seems Sports and Tech videos are easy to distinguish from remaining genres.

## 7.7 Issues

Many named entities in the Youtube media, especially unigrams, are ambiguous. Incorrect concept definition retrieval from the Wikipedia, arising out of ambiguity may inject noise into the system or can be ignored. For Example, a Sports video with the title "*Manchester rocks*" refers to the *Manchester United Football Club*. But Wikipedia returns a general article on the *city of Manchester* in England. None of the words in its definition matches any word in seed word lists and the entity is ignored.

Considering only WordNet synsets gives less coverage. Considering the gloss information helps to some extent. For example, if the word "*shot*" is not present in the seed list for *Sports*, then "*dunk*" cannot be associated to the *Sports* genre. But this association can be properly captured through the gloss of the WordNet first sense of "*dunk*" (- *a basketball shot in which the basketball is propelled downward into the basket*). However, it runs the risk of incorporating noise. Consider the word *good* and the gloss of one of its synsets {*dear, good, near -- with or in a close or intimate relationship*}. Here the word "*good*" is associated to *Romance* due to the presence of "*relationship*", which is incorrect.

Uploader provided video meta-data is typically small and require concept expansion to extract useful information. User comments provide a lot of information but incorporate noise as well. Auto-generated bot advertisements for products, off-topic conversation between users, fake urls, mis-spelt words, different forms of slangs and abbreviations mar the accuracy. For example, an important seed word for the *Romance* genre will not be recognized if "*love*" is spelt as "*luv*", which is common.

# 8 CONCLUSION

In this work, we propose a weakly supervised system, *YouCat*, for predicting *possible genre tags* for a video using the *video title*, *meta description* and the *user comments*. *Wikipedia* and *WordNet* are used for expanding the extracted concepts to detect cue words from a genre-specific seed set of words. The weak supervision arises out of the usage of a root words list (~ 1-3 words) used to describe the genre, usage of WordNet which is manually tagged and the simple parameter setting for the model. There are a number of parameters which have been simplistically set. Tuning the parameters using labeled data may improve the accuracy. An accuracy of *80.9%* in single genre prediction and *91.48%* in multiple genre prediction is obtained without using *any labeled data,* compared to the supervised multi-class SVM baseline of *84.36%* in single genre prediction. The accuracy suffers due to the inherent noise in the Youtube media arising out of the user comments and incorrect concept expansion due to ambiguity. A pre-processing filter that allows only relevant user comments about the video and a WSD module will boost the performance of the system. This work is significant as it does not use any manually labeled data for training and can be automatically extended for multiple genres with minimal supervision. This work also exhibits the usefulness of user information and concept expansion though WordNet and Wikipedia in video categorization.

## References

1. Bishop, Christopher M. Pattern Recognition and Machine Learning (Information Science and Statistics). 2006. Springer-Verlag New York, Inc.

2. Borth, Damian and Hees, J\"{o}rn and Koch, Markus and Ulges, Adrian and Schulze, Christian and Breuel, Thomas and Paredes, Roberto, TubeFiler – an Automatic Web Video Categorizer, Proceedings of the 17th ACM international conference on Multimedia, MM '09

3. Cui, Bin and Zhang, Ce and Cong, Gao, Content-enriched classifier for web video classification, Proceedings of the 33rd international ACM SIGIR, 2010

4. Ekenel, Hazim Kemal and Semela, Tomas and Stiefelhagen, Rainer, Content-based video genre classification using multiple cues, Proceedings of the 3rd international workshop on Automated information extraction in media production, AIEMPro '10

5. Filippova, Katja and Hall, Keith B., Improved Video Categorization from Text Metadata and User Comments, Proceedings of the 34th international ACM SIGIR, 2011, pp. 835-842

6. Giuliano Armano and Alessandro Giuliani and Eloisa Vargiu, Experimenting Text Summarization Techniques for Contextual Advertising, Proceedings of the 2nd Italian Information Retrieval (IIR) Workshop, Milan, Italy, 2011

7. Huang, Chunneng and Fu, Tianjun and Chen, Hsinchun, Text-Based Video Content Classification for Online Video-Sharing Sites, Journal of the American Society for Information Science and Technology Volume 61, Issue 5, pages 891–906, 2010

8. Macdonald, Craig and Ounis, Iadh, Expertise drift and query expansion in expert search, Proceedings of the sixteenth ACM conference on Conference on information and knowledge

management, CIKM '07, 2007

9. Manning, Christopher D. and Raghavan, Prabhakar and Schtze, Hinrich. Introduction to Information Retrieval. Cambridge University Press, 2008

10. McCarthy, Diana and Koeling, Rob and Weeds, Julie and Carroll, John, Finding Predominant Word Senses in Untagged Text, Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), 2004

11. S. Zanetti, L. Zelnik-Manor, and P. Perona, A walk through the web's video clips, In Proc. of CVPR Workshop on Internet Vision, 2008.

12. Song, Yicheng and Zhang, Yong-dong and Zhang, Xu and Cao, Juan and Li, Jing-Tao, Google Challenge: Incremental-Learning for Web Video Categorization on Robust Semantic Feature Space, Proceeding MM '09 Proceedings of the 17th ACM International conference on Multimedia, 2009

13. Wu, Xiao and Ngo, Chong-Wah and Zhu, Yi-Ming and Peng, Qiang, Boosting web video categorization with contextual information from social web, World Wide Web Volume 15 Issue 2, 2012

14. Yang, Linjun and Liu, Jiemin and Yang, Xiaokang and Hua, Xian-Sheng, Multi-Modality Web Video Categorization, Proceeding MIR '07 Proceedings of the international workshop on Workshop on multimedia information retrieval, 2007

15. Yew, Jude and Shamma, David A. and Churchill, Elizabeth F., Knowing funny: genre perception and categorization in social video sharing. In Proceedings of CHI'2011. pp.297-306

16. Zhang, John R. and Song, Yang and Leung, Thomas, Improving Video Classification via YouTube Video Co-Watch Data, ACM Workshop on Social and Behavioural Network Media Access at ACM MM 2011

17. Zhang, Ruofei and Sarukkai, Ramesh and Chow, Jyh-Herng and Dai, Wei and Zhang, Zhongfei, Joint Categorization of Queries and Clips for Web-based Video Search, Proceeding MIR '06 Proceedings of the 8th ACM international workshop on Multimedia information retrieval, 2006

18. Zheshen Wang, Ming Zhao, Yang Song, Sanjiv Kumar, and Baoxin Li, YouTubeCat: Learning to Categorize Wild Web Videos, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2010.