# Previsão de Popularidade de Vídeos no YouTube Utilizando Padrões de Acesso Iniciais

**Henrique Pinto**[1], **Jussara Almeida**[1], **Marcos Gonçalves**[1]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais

{hpinto,jussara,mgoncalv}@dcc.ufmg.br

***Abstract.*** *Predicting the popularity of content on the Web is an important task, with many potential uses, spanning from the design of efficient cache systems to evaluating the performance of recommendation services. We present a simple model for predicting the future popularity of Web content based on the historical information given by early popularity measures. Our approach is validated on datasets consisting of videos from the well-known YouTube video-sharing portal. Our experimental results show that we can obtain significant reductions (up to 69%) in relative squared errors when compared to a state-of-the-art baseline model, especially on videos that have a very high peak in views followed by a sudden large decrease in popularity.*

***Resumo.*** *Previsão de popularidade de conteúdo na Web é uma tarefa importante, com aplicações potenciais que vão do projeto de sistemas de cache eficientes à avaliação de performance de serviços de recomendação. Apresentamos um modelo simples para prever a popularidade futura de conteúdo na Web com base em informações históricas dadas por medições de popularidade no começo da vida do conteúdo. Nossa abordagem é validada em duas bases de dados do YouTube, um portal de compartilhamento de vídeo amplamente conhecido. Os resultados experimentais mostram que é possível obter reduções significativas (de até* $69\%$*) no erro relativo quadrático médio quando comparado a um modelo estado-da-arte, em especial em vídeos cuja curva de popularidade é caracterizada por um pico significativo seguido de uma queda brusca.*

## 1. Introdução

A crescente popularidade de aplicações *Web 2.0* trouxe uma grande quantidade de conteúdo gerado por usuários. Considere, por exemplo, o sistema de compatilhamento de vídeos YouTube, que frequentemente fica entre as três aplicações mais populares na Web [Alexa.com 2011]: usuários do YouTube fazem *upload* de quase 48 horas de vídeo por *minuto*[1], e a quantidade total de conteúdo adicionado ao sistema em 60 dias é equivalente a todo o conteúdo transmitido nos últimos 60 anos, sem interrupção, pelas redes de televisão NBC, CBS e ABC juntas [Heffernan 2009]. Dada essa alta taxa de criação de conteúdo, não é surpreendente que a distribuição de popularidade de conteúdo na Web 2.0, e no Youtube em particular, seja muito desigual: a maioria do conteúdo recebe bem pouca atenção enquanto que uma pequena parte atrai milhões de visualizações [Cha et al. 2007, Wu and Huberman 2007].

[1]http://www.youtube.com/t/press_statistics

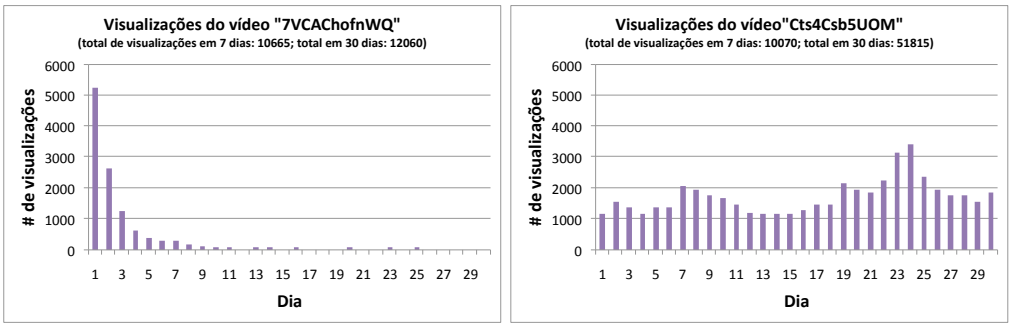
Nesse contexto, prever a popularidade de conteúdo passa a ser uma tarefa importante para suportar o projeto e gerenciamento de diversos serviços. Por exemplo, um administrador de sistemas pode utilizar previsões de popularidade futura para planejar pró-ativamente atualizações de infraestrutura ou para contrtar serviços de terceiros, como CDNs. Previsão de popularidade de conteúdo também é fundamental para apoiar estratégias de marketing online e marketing viral, bem como serviços de busca e recomendação de conteúdo eficazes [Cha et al. 2009, Gonçalves et al. 2010]. Por exemplo, a previsão pode ajudar a identificar possíveis gargalos devido à resultados ruins de busca ou recomedação, e pode também ser usada para melhorar a qualidade desses serviços, extendendo as estratégias de *ranking* para levar em consideração a popularidade futura [Gonçalves et al. 2010].

Há esforços recentes no sentido de criar modelos para prever a popularidade de conteúdo *on-line* usando várias técnicas, tais como *reservoir computing* [Wu et al. 2010], modelos estocásticos de comportamento de usuário [Lerman and Hogg 2010] e técnicas de análise de sobrevivência [Lee et al. 2010]. Szabo e Huberman observaram que a popularidade a longo prazo $\log$-transformada de um conteúdo é fortemente correlacionada com sua popularidade nos primeiros momentos após ele ser adicionado ao sistema[Szabo and Huberman 2010]. Baseado nessa observação, eles propuseram um modelo simples que prevê que a popularidade (em termos de número de visualizações) de um conteúdo em uma data-alvo $t_s$ é dada pelo produto entre uma constante $\alpha$ e o número de visualizações em uma data de referência $t_r$ ($t_r < t_s$). Essa constante depende apenas das datas de referência e alvo e não de nenhuma outra informação particular do conteúdo, e pode ser calculada por regressão linear simples.

O modelo de Szabo e Huberman (modelo S-H) produz resultados bons, em especial dada sua simplicidade. Ele, porém, possui limitações. Em particular, dois conteúdos diferentes podem exibir popularidade (medida em número de visualizações) similar na data de referência, mas apresentar comportamentos extremamente distintos. Considere, por exemplo, as curvas de popularidade de dois vídeos reais do YouTube que são mostradas na Figura 1. Sete dias após serem adicionados ao sistema, ambos os vídeos possuem aproximadamente o mesmo número de visualizações ($10.665$ para o primeiro, $10.070$ para o segundo). Portanto, o modelo S-H, considerando a data de referência como sendo 7 dias e a data alvo como sendo 30 dias, iria prever que ambos os vídeos teriam aproximadamente a mesma popularidade em 30 dias. No entando, eles terminam com popularidades *muito* diferentes: enquanto o vídeo da Figura 1-(a) mal ultrapassa $12.000$ visualizações na data alvo, o vídeo da Figura 1-(b) acaba atraindo mais de $50.000$ visualizações no mesmo período.

Analisando a Figura 1, porém, fica claro que os dois vídeos *devem* ter popularidades futuras *muito* diferentes. Ainda que eles tenham aproximadamente o mesmo número de visualizações após sete dias, um deles foi muito popular no primeiro dia e viu sua popularidade cair rapidamente, enquanto que o outro não apresenta nenhuma tendência específica (apesar de algumas flutuações). De fato, Crane e Sornette recentemente distinguiram quatro classes diferentes de padrões de evolução de popularidade em vídeos do YouTube, que são explicadas em termos de uma combinação de fatores endógenos (i.e., interações dentro do próprio sistema) e exógenos (eventos externos) [Crane and Sornette 2008], como discutiremos na Seção 4.

Portanto, investigamos aqui se levar em conta os *padrões iniciais de popularidade*




(a) (b)

**Figura 1. # de visualizações diárias nos primeiros 30 dias, para dois vídeos**

*dade* que um vídeo exibe durante seus primeiros dias no sistema pode ajudar a criar previsões mais precisas de seu número total de visualizações em uma dada data alvo, em comparação com a hipótese mais simples de levar em conta apenas o número total de visualizações até a data de referência. Em outras palavras, propomos um modelo de regressão linear que estende o modelo S-H de forma tal que ele possa considerar o padrão de acesso ao vídeo em seus primeiros dias. Ele é motivado pela observação principal de que se observamos o número de visualizações até uma certa data de referência de $t_r$ dias para prever a popularidade futura, nem todos os dias são igualmente importantes. Alguns vídeos terão picos de popularidade logo nos primeiros dias e então uma queda drástica em seu número de visualizações por dia. Outros vídeos manterão um número estável de visualizações por dia, com pequenas variações, mas sem picos destacados nem quedas bruscas. Assim, ainda prevemos a popularidade futura baseado unicamente no número de visualizações que um vídeo recebe até a data de referência $t_r$, mas atribuímos pesos diferentes para cada dia entre o *upload* do vídeo e o dia de referência. Esses pesos permitem diferenciar vídeos como os mostrados na Figura 1, mesmo quando eles possuem aproximadamente o mesmo número total de visualizações até a data de referência.

Dada a diversidade de conteúdo e os vários fatores que podem impactar a popularidade de vídeos, nós avaliamos nosso modelo, comparado ao modelo S-H, em duas bases de dados de vídeos do YouTube: (1) *top*, que consiste de vídeos populares que apareceram nas listas de *top* vídeos mundiais que o sistema disponibiliza; e (2) *random*, que contém vídeos amostrados de acordo com um procedimento aleatório [Figueiredo et al. 2011]. Para cada base de dados, medimos quão bom é o desempenho de cada modelo, em termos de erro de previsão, para vários valores de datas de referência e alvo, bem como para vídeos que exibem diferentes padrões de evolução de popularidade (tais como definidos em [Crane and Sornette 2008]). Nosso modelo melhora significativamente os resultados do modelo S-H, reduzindo o erro de previsão, em média, por $13\%$ e $15\%$ para vídeos nos datasets *random* e *top*, respectivamente. Além disso, a redução no erro pode chegar a $36\%$ e $69\%$ nas bases de dados *top* e *random*, respectivamente, para vídeos que apresentam picos significativos de popularidade seguidos por quedas bruscas (como na Figura 1-(b)).

Analisamos ainda se criar modelos específicos para diferentes tipos de vídeos pode levar a previsões melhores. Por exemplo, consideramos construir modelos de previsão diferentes para cada categoria de vídeos no YouTube. No entanto, exceto para duas categorias (uma em cada base de dados), especialização de modelos produz pouco impacto no erro, principalmente porque vídeos na maioria das categorias seguem o mesmo padrão

geral de popularidade.

O restante do artigo é organizado da seguinte maneira: primeiramente, revisamos trabalhos relacionados (Seção 2); em seguida, apresentamos o modelo S-H e nosso modelo na Seção 3. A metodologia de avaliação e principais resultados são discutidos nas Seções 5 e 6, respectivamente. Por fim, apresentamos nossas conclusões na Seção 7.

## 2. Trabalhos Relacionados

Entender (e prever) a popularidade de conteúdo na Web é um tópico bastante estudado ultimamente. Cha *et al.* fizeram um estudo detalhado do ciclo de popularidade de vídeos no YouTube [Cha et al. 2007]. Eles observaram que vídeos no sistema possuem, em média, tempos de vida longos: cerca de $80\%$ dos vídeos requisitados em um dia são mais velhos do que um mês. Apesar disso, o vídeo individual mais visto em um dado dia tende a ser um vídeo novo. Figueiredo *et al.* caracterizaram os padrões de crescimento de popularidade de vídeos no YouTube, usando informações dos *referrers* dos vídeos [Figueiredo et al. 2011], e mostraram que busca e mecanismos internos do YouTube (como vídeos relacionados) são os principais responsáveis por atrair usuários para vídeos. Já Rodrigues *et al.* analisaram a popularidade de cópias duplicadas de vídeos no YouTube [Rodrigues et al. 2010]. Eles observaram que diferentes cópias de um *mesmo* vídeo podem possuir padrões de popularidade extremamente diferentes, e concluem que o usuário que postou o vídeo é um fator importante para determinar sua popularidade.

Szabo e Huberman analisaram a popularidade de vídeos no YouTube e de histórias no portal de agregação de conteúdo Digg [2], notando que a popularidade a longo prazo de um conteúdo é fortemente correlacionada com a popularidade a curto prazo numa escala logarítmica. Com base nessa observação, propuseram o modelo de previsão simples que usaremos como linha de base e que descrevemos com mais detalhes na Seção 3.2.

Outros modelos para previsão de popularidade já foram propostos. Lerman e Hogg modelam o comportamento do usuário explicitamente, como um processo estocástico, e com base nisso criam um modelo de previsão para histórias no Digg [Lerman and Hogg 2010]. O modelo produz bons resultados, mas é de aplicação limitada em um sistema como o YouTube, dado que o YouTube possui muitos outros recursos e mecanismos internos, que permitem uma maior variedade de interações entre usuários, que ultimamente impactam a popularidade. Assim, modelar o comportamento de usuário nesse sistema é uma tarefa mais complexa, em especial porque a popularidade de conteúdo é muito influenciada por fatores externos, como busca e *links* externos.

Wu *et al.* usam um modelo baseado em *reservoir computing* (um tipo de rede neural) para previsão de popularidade a curto prazo de vídeos [Wu et al. 2010]. A estratégia, porém, não produz ganhos em relação à estratégias mais simples em bases de dados grandes, e pode ser afetada por randomização, dado que os pesos da camada escondida são configurados como valores aleatórios *fixos*, de forma que o método necessita de algumas iterações até que a saída reflita os dados em vez dos pesos aleatórios.

Lee *et al.* propuseram um modelo para inferir a probabilidade de que um certo conteúdo ainda será popular em um dado ponto de tempo no futuro [Lee et al. 2010]. Usando técnicas de *análise de sobrevivência* emprestadas da biologia, eles criam um modelo que busca prever o tempo de vida de uma *thread* em fóruns de discussão com base em

[2] `http://digg.com`

alguns *fatores de risco*, como o número de comentários e o tempo entre a criação da thread e o primeiro comentário. Esse modelo pressupõe que medições diretas de popularidade não estão disponíveis, o que pode ser muito restritivo num sistema como o YouTube, que disponibiliza essa informação publicamente. Além disso, verificamos que os fatores de risco utilizados são altamente correlacionados com o número de visualizações dos vídeos em nossas bases de dados, e não trouxeram nenhum ganho para nosso modelo.

## 3. Modelos de Previsão de Popularidade

Um bom modelo de previsão é aquele que produz previsões precisas, para alguma definição de "precisão". Assim, inicialmente discutimos o critério usado para avaliar a performance dos modelos de previsão usados (Seção 3.1). Com esse critério definido, introduzimos formalmente o modelo S-H, que trataremos como linha de base (Seção 3.2), e nosso novo modelo linear multivariado (Seção 3.3), discutindo como ambos otimizam o critério de performance adotado.

No que segue, definimos a popularidade de um vídeo $v$ após $t_t$ dias de sua entrada no sistema como o *número total de visualizações* que ele recebeu até essa data. O objetivo dos modelos de previsão é estimar a popularidade futura em uma data $t_t$, após observar o vídeo até uma certa data de referência $t_r$ ($t_r < t_t$).

### 3.1. Critério de Performance

Usamos como critério de avaliação o Erro Quadrático Relativo (RSE). Seja $N(v, t_t)$ o número total de visualizações que o vídeo $v$ recebe até a data $t_t$, e $\hat{N}(v, t_r, t_t)$ o número *previsto* de visualizações para o vídeo $v$ na data $t_t$, usando como dados para a previsão informações coletadas até a data $t_r$. O RSE para essa previsão é definido como:

$$RSE = \left( \frac{\hat{N}(v, t_r, t_t)}{N(v, t_t)} - 1 \right)^2 \tag{1}$$

Para uma coleção $C$ de vídeos, definimos o Erro Quadrático Relativo Médio (mRSE) como a média aritmética do RSE para todos os vídeos na coleção:

$$mRSE = \frac{1}{|C|} \cdot \sum_{v \in C} \left( \frac{\hat{N}(v, t_r, t_t)}{N(v, t_t)} - 1 \right)^2 \tag{2}$$

A razão para usar o RSE em vez de outras métricas mais tradicionais, como o erro quadrático (não-relativo) é que para os vários serviços para os quais a previsão de popularidade é útil, erros relativos são mais relevantes que os erros absolutos, em particular dada a grande variação da popularidade entre diferentes vídeos no YouTube [Cha et al. 2007]. Por exemplo, um erro de previsão de $5\%$ em um vídeo de milhões de visualizações seria *muito grande* em termos absolutos, mas pequeno o suficiente para que ele não faça diferença significativa em, digamos, uma política de cache. Da mesma forma, um erro de $1000\%$ em um vídeo de poucas centenas de visualizações seria pequeno em termos absolutos (comparado com o erro de $5\%$ do cenário anterior), mas alto em termos relativos.

O RSE não é definido para vídeos que possuem zero visualizações na data alvo. Em nossas bases de dados, porém, observamos que apenas uma fração muito pequena de

vídeos não recebe nenhuma visualização até alguma das datas alvos que consideramos em nossos experimentos: menos de $1,5\%$ dos vídeos possuem zero visualizações em 30 dias após o *upload*. Assim sendo, ignoramos esses vídeos no cálculo do mRSE.

### 3.2. Modelo de Szabo-Huberman (S-H)

Szabo e Huberman observaram uma correlação linear forte entre as popularidades inicial e futura, quando log-transformadas, com um ruído de distribuição normal. Baseado nessa observação, eles expressaram a popularidade futura de um dado conteúdo $v$ como:

$$\hat{N}(v, t_r, t_t) = \alpha_{t_r,t_t} \cdot N(v, t_r) \tag{3}$$

onde o parâmetro $\alpha_{t_r,t_t}$ é independente de $v$. Para datas alvo ($t_t$) e de referência ($t_r$) fixas, o modelo prevê que a popularidade futura de $v$ é relacionada com sua popularidade inicial por um fator constante.

Para um dado par $(t_r, t_t)$, podemos calcular o valor ótimo para $\alpha_{t_r,t_t}$ em um conjunto de treino $C$ substituindo o valor de $\hat{N}(v, t_r, t_t)$ na Equação 2 pelo hipótese do modelo (Equação 3), calculando sua derivada e igualando-a a zero. Isso nos leva a:

$$\alpha_{t_r,t_t} = \frac{\sum_{v \in C} \frac{N(v,t_r)}{N(v,t_t)}}{\sum_{v \in C} \left(\frac{N(v,t_r)}{N(v,t_t)}\right)^2}$$

### 3.3. Modelo Linear Multivariado (ML)

O modelo S-H usa como entrada o número total de visualizações que um vídeo recebeu até $t_r$. Considere que em vez de ter acesso apenas a esse número, nós amostrássemos o número de visualizações em intervalos regulares até a mesma data de referência. Por simplicidade, no que segue consideraremos que o intervalo de amostragem é de um dia, mas o modelo em si não faz nenhuma suposição sobre ele.

Agora, em vez de usar apenas a popularidade agregada em $t_r$, podemos usar os *deltas* entre cada amostra para prever a popularidade em $t_t$. Por exemplo, se a data de referência é $t_r = 7$ dias, e a amostragem é diária, teríamos sete valores, que correspondem ao número de views recebidos por dia para cada um dos primeiros sete dias.

Consideremos, então, um modelo onde a popularidade prevista em $t_t$ é uma função linear desses deltas. A suposição de linearidade vem da forte correlação linear entre popularidades iniciais e futuras observadas por Szabo e Huberman [Szabo and Huberman 2010]. Ainda assim, o modelo é mais poderoso que o modelo S-H porque ele permite associar diferentes "pesos" a cada intervalo de amostragem. Por exemplo, considerando os vídeos da Figura 1. Ambos possuem aproximadamente o mesmo número total de views em $t_r = 7$, mas é possível diferenciar entre eles facilmente, atribuindo pesos menores para os dias iniciais.

Mais formalmente, seja $x_i(v)$ o número de visualizações recebidas por um vídeo $v$ no $i$-ésimo dia desde seu upload[3]. O *vetor de atributos* $X_{t_r}(v)$ é definido como:

$$X_{t_r}(v) = (x_1(v),\, x_2(v),\, \cdots,\, x_{t_r}(v))^T$$

[3] $x_i(v) = N(v, i) - N(v, i-1)$, onde $N(v, i)$ é o número total de visualizações até o $i$-ésimo dia e $N(v, 0) = 0$

e estimamos a popularidade de um vídeo $v$ na data $t_t$ como:

$$\hat{N}(v, t_r, t_t) = \Theta_{(t_r,t_t)}^T \cdot X_{t_r}(v) \tag{4}$$

onde $\Theta_{(t_r,t_t)} = (\theta_1, \theta_2, \cdots, \theta_{t_r})^T$ é o *vetor de parâmetros do modelo* e, assim como o parâmetro $\alpha$ do modelo S-H, depende apenas das datas de referência e alvo, mas não de nenhuma informação do vídeo em si.

Dado um conjunto de treino $C$ e datas de referência e alvo $t_r$ and $t_t$, podemos calcular o valor ótimo para $\Theta_{(t_r,t_t)}$ como aquele que minimiza o mRSE em $C$. Para tal, primeiro nós substituímos o valor de $\hat{N}(v, t_r, t_t)$ na Equação 2 pelo valor dado pela Equação 4, e então calculamos as derivadas parciais em relação a $\theta_1$, $\theta_2$, ..., $\theta_{t_r}$, o que nos leva a:

$$\frac{\partial mRSE}{\partial \theta_i} = \frac{2}{|C|} \sum_{v \in C} \left[ \left( \frac{\Theta_{(t_r,t_t)}^T \cdot X_{t_r}(v)}{N(v,t)} - 1 \right) \cdot \frac{x_i(v)}{N(v,t)} \right]$$

Igualando todas as derivadas parciais à zero e resolvendo o sistema linear resultante para $\Theta$ nos dá os valores ótimos para os parâmetros. Alternativamente, é possível usar algum algoritmo de otimização tal como o BFGS de memória limitada [Liu and Nocedal 1989] para calcular iterativamente os valores ótimos, o que pode ser mais eficiente se o número de parâmetros é grande. Esse cálculo é necessário apenas numa fase inicial de treinamento. Uma vez que isso tenha sido feito, o modelo permite realizar previsões de forma bem rápida, dado que ele requer apenas $t_r$ multiplicações e uma soma final de $t_r$ termos.

É interessante notar que o modelo S-H é um caso especial desse modelo multivariado, com a restrição adicional de que $\theta_1 = \theta_2 = \cdots = \theta_{t_r}$. Assim, exceto se houver *overfit*, espera-se que os resultados produzidos por esse sejam iguais-ou-melhores aos do S-H. Uma possível desvantagem de nosso modelo, porém, é que o número de parâmetros não é fixo, mas cresce linearmente com $t_r$. Acreditamos que isso não representa um problema, na prática, por duas razões: (1) geralmente, quer-se estimar a popularidade o mais cedo possível, e uma data de referência pequena implica em poucos parâmetros; e (2) se o número total de amostras (e, portanto, de parâmetros) é grande demais, podemos reduzi-lo usando um intervalo amostral maior (por exemplo, semanal em vez de diário).

## 4. Padrões de Evolução de Popularidade

Como a Figura 1 mostra, claramente há padrões de evolução de popularidade muito diferentes que vídeos podem seguir. Crane e Sornette analisaram milhões de vídeos do YouTube e identificaram quatro classes principais de comportamento de popularidade, que podem ser explicadas em termos de efeitos endógenos e exógenos e da habilidade de usuários influenciarem outros [Crane and Sornette 2008]:

- *Junk*: vídeos na classe *Junk* experimentam um pico súbito de popularidade causado por algum efeito externo, mas sua popularidade decai rapidamente após esse pico pois os usuários não o propagam pela rede social.
- *Quality*: esses vídeos, assim como os Junk, experimentam um pico súbito de popularidade devido a um efeito externo. Porém, a popularidade decai mais lentamente, dado que os usuários iniciais propagam o vídeo para outros usuários, e assim por diante.
- *Viral*: nesses vídeos, não há um efeito externo significativo que causa o pico de popularidade. O vídeo se espalha na rede social, com sua popularidade crescendo lentamente até atingir um pico, a partir do qual ela decresce lentamente.

• *Memoryless*: por fim, há vídeos que não experimentam picos significativos de popularidade, e a variação é dada por flutuações que podem ser explicadas por um processo de Poisson.

Nos experimentos, analisamos o comportamento dos modelos em relação a cada uma dessas classes. Os vídeos nas Figuras 1-(a) e 1-(b) estão, respectivamente, nas classes *Junk* e *Memoryless*.

## 5. Metodologia

Nossos experimentos foram realizados usando duas bases de dados de vídeos do YouTube:

• Base *top*, que contém vídeos coletados das *Top Lists* globais do sistema. A base de dados original é composta de $27212$ vídeos, mas após aplicar filtros para remover vídeos sem informação de popularidade (ou com informação inconsistente), vídeos com menos de 30 dias de vida e vídeos com zero visualizações após 30 dias, restaram $5834$ vídeos.
• Base *random*, que contém vídeos amostrados segundo um processo aleatório. Palavras-chave foram escolhidas aleatoriamente e usadas como consultas para buscas no YouTube. Os vídeos retornados por essas buscas compõe essa base, que contém $24484$ vídeos ($16123$ após os filtros).

Em ambos os casos, a informação de popularidade foi extraída do quadro de estatísticas do vídeo fornecido pelo YouTube, onde ela aparece em forma de gráfico. Esse gráfico é plotado usando a API do Google Charts usando 100 pontos. Foi possível interceptar essa chamada e obter esses pontos. Note que para vídeos com mais de 100 dias de vida, não há informação sobre todos os dias. Nesse caso, foi usada interpolação linear entre os pontos dados para inferir os dados ausentes. Essa interpolação pode, obviamente, impactar os resultados. Fizemos experimentos usando apenas vídeos com menos de 100 dias (para os quais ela não ocorre) e observamos que os resultados qualitativos se mantém, e que em termos quantitativos o erro de previsão ao usar dados interpolados é *maior* que o erro obtido usando apenas vídeos sem interpolação. Optamos por usar as bases completas devido ao tamanho delas – há relativamente poucos vídeos nas bases que foram coletados com entre 30 e 100 dias de vida (e para os quais, portanto, não há interpolação).

Todos os experimentos foram realizados com validação cruzada de 10 *folds*. Os resultados apresentados são a média e intervalo de confiança de $95\%$ dos resultados dos 10 folds.

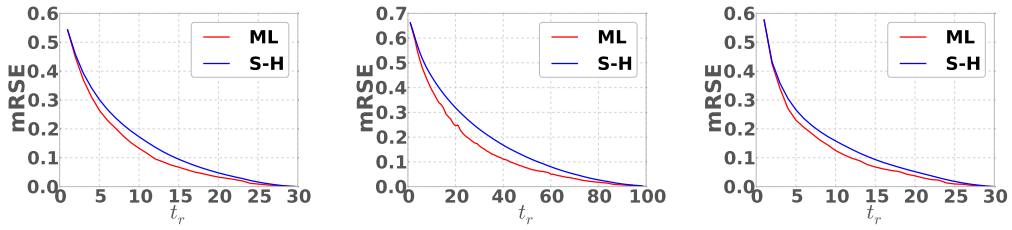
## 6. Resultados

Discutimos aqui os principais resultados experimentais para ambos os modelos. Inicialmente, apresentamos os resultados produzidos por cada modelo em ambas as bases de dados (Seção 6.1). Então, investigamos se há benefício em construir modelos especializados por categoria ou classe de popularidade (Seção 6.2).

### 6.1. Considerar os padrões de acesso iniciais melhora a previsão?

A Figura 2-(a) mostra os valores de mRSE para ambos os modelos ao prever a popularidade de vídeos na base de dados *random*, com data alvo $t_t = 30$ dias, para vários valores de data de referência $t_r$ (i.e., o número de dias usados para a previsão). Se $t_r$ é muito pequeno, há pouca informação extra da qual o modelo ML pode se beneficiar, e assim ambos

os modelos têm desempenho semelhante. No outro extremo, se quase toda a informação anterior à data alvo está disponível (i.e., se $t_r$ é próximo de $t_t$), então os erros são tão pequenos em valor absoluto que há pouco espaço para melhoras significativas. Prever a popularidade nesse caso, porém, tem pouca aplicação prática. Porém, a Figura 2-(a) mostra que, fora desses extremos, o modelo ML pode levar a reduções significativas de mRSE comparado ao modelo S-H, com ganhos de mais de $20\%$. Por exemplo, se $t_r = 12$ dias, o mRSE dos modelos ML e S-H é de, respectivamente, $0.081 \pm 0.009$ and $0.104 \pm 0.008$, o que representa melhora de $22\%$. Comportamento semelhante é observado tanto quando mudamos a base de dados (veja Figura 2-(c), que mostra os resultados para a base *top* com mesma data alvo) e quando mudamos a data alvo (a Figura 2-(b) mostra os resultados para a base *random* com $t_t = 100$ dias). Nesse último caso, a redução de erro chega a ser consideravelmente maior: para $t_t = 100$ dias e $t_r = 40$ dias, o modelo ML produz mRSE de $0.1112 \pm 0.0024$, uma redução de $33\%$ em relação ao mRSE de $0.1679 \pm 0.0025$ produzido pelo modelo S-H.




(a) $t_t$= 30 dias, base *random* (b) $t_t$ =100 dias, base *random* (c) $t_t$= 30 dias, base *top*

**Figura 2. Valores de mRSE como uma função da data de referência $t_r$ para várias datas alvo $t_t$**

Para melhor entender a performance de nosso modelo, analisamos separadamente os valores de mRSE produzidos por ambos os modelos para vídeos em vários padrões distintos de evolução de popularidade. Nós consideramos as quatro classes de popularidade identificadas por Crane e Sornette [Crane and Sornette 2008]: *Memoryless*, *Viral*, *Junk* e *Quality*, como apresentadas na Seção 4.

A Tabela 1 mostra os resultados de mRSE produzidos por ambos os modelos considerando $t_r = 7$ e $t_t = 30$ em ambas as bases de dados. Resultados agregados e por classe de popularidade são mostrados, e os resultados do modelo mais preciso em cada caso são mostrados em negrito (incluindo empates estatísticos)[4] As equações dos modelos, mostrando os valores ótimos encontrados durante o treino, também são mostradas. Os erros produzidos por nosso modelo foram, no geral, $15\%$ menores na base *random* e $13\%$ menores na base *top*. Além disso, nosso modelo produz erros significativamente menores na maioria das classes, com alguns empates estatísticos nas demais. Os ganhos são especialmente grandes para vídeos na classe *Junk*, chegando a exceder $69\%$ na base de dados *random*.

Vídeos da classe *Junk* são caracterizados por um pico súbito de popularidade, que concentra uma parcela muito grande do total de visualizações, seguido de uma queda brusca. Através de algum mecanismo externo, esses vídeos atingem alta popularidade

[4]Note que, apesar de resultados por classe serem apresentados, apenas um modelo global S-H e um modelo global ML foram construídos a partir do conjunto de treino.

em algum momento, mas eles não são *interessantes* o suficiente para continuar sendo populares a longo termo. Uma análise simples mostrou que esses vídeos, em geral, experimentam o seu pico de popularidade muito cedo: em ambas as bases de dados, cerca de $90\%$ dos vídeos dessa classe têm seu pico em um dos três primeiros dias de vida. Nosso modelo se adequa a esses vídeos atribuindo pesos menores aos primeiros dias (veja a equação dos modelos na Tabela 1) e, assim, produz resultados muito melhores para eles que o modelo S-H, que tende a superestimar a popularidade deles. Em outras palavras, mesmo que um vídeo receba muitas visualizações nos primeiros dias, se ele já não recebe quase nenhuma visualização, digamos, no sétimo dia, ele provavelmente não vai se tornar muito mais popular no futuro. Na base de dados *top*, o mRSE para vídeos *Junk* já era muito baixo, e não houve espaço para uma redução estatisticamente significativa.

**Tabela 1. Erro de previsão produzido pelos modelos para vídeos de diferentes classes de evolução de popularidade (mRSE, $t_r = 7$, $t_t = 30$).**

| **Vídeos na base de dados *random*** | | | | |
|---|---|---|---|---|
| **Classe de Popularidade** | **# de Vídeos** | **Modelo S-H** | **Nosso Modelo (ML)** | **Dif. %** |
| Geral | 16123 | $0.2382 \pm 0.0038$ | $\mathbf{0.2022 \pm 0.0043}$ | $-15.1\%$ |
| Memoryless | 3449 | $0.3351 \pm 0.0099$ | $\mathbf{0.2929 \pm 0.0120}$ | $-12.6\%$ |
| Viral | 11504 | $0.2086 \pm 0.0040$ | $\mathbf{0.1788 \pm 0.0041}$ | $-14.3\%$ |
| Junk | 222 | $0.4588 \pm 0.0283$ | $\mathbf{0.1402 \pm 0.0274}$ | $-69.4\%$ |
| Quality | 948 | $\mathbf{0.1921 \pm 0.0124}$ | $\mathbf{0.1707 \pm 0.0283}$ | $-11.2\%$ |
| **Vídeos na base de dados *top*** | | | | |
| **Classe de Popularidade** | **# de Vídeos** | **Modelo S-H** | **Nosso Modelo (ML)** | **Dif. %** |
| Geral | 5813 | $0.2121 \pm 0.0074$ | $\mathbf{0.1837 \pm 0.0081}$ | $-13.4\%$ |
| Memoryless | 5130 | $0.2104 \pm 0.0081$ | $\mathbf{0.1820 \pm 0.0088}$ | $-13.5\%$ |
| Viral | 401 | $0.3096 \pm 0.0234$ | $\mathbf{0.2612 \pm 0.0228}$ | $-15.6\%$ |
| Junk | 78 | $\mathbf{0.0560 \pm 0.0226}$ | $\mathbf{0.0360 \pm 0.0236}$ | $-35.7\%$ |
| Quality | 204 | $\mathbf{0.1236 \pm 0.0314}$ | $\mathbf{0.1315 \pm 0.0330}$ | $6.4\%$ |

**Equações dos modelos, base de dados *random*:**

Modelo S-H: $\hat{N}(v, 7, 30) = 1.92 \cdot N(v, 7)$

Modelo ML: $\hat{N}(v, 7, 30) = 1.22 \cdot x_1(v) + 1.24 \cdot x_2(v) + 1.36 \cdot x_3(v) + 1.52 \cdot x_4(v) + 2.23 \cdot x_5(v) + 2.33 \cdot x_6(v) + 6.15 \cdot x_7(v)$

**Equações dos modelos, base de dados *top*:**

Modelo S-H: $\hat{N}(v, 7, 30) = 1.41 \cdot N(v, 7)$

Modelo ML: $\hat{N}(v, 7, 30) = 1.19 \cdot x_1(v) + 1.02 \cdot x_2(v) + 1.16 \cdot x_3(v) + 1.36 \cdot x_4(v) + 1.35 \cdot x_5(v) + 1.47 \cdot x_6(v) + 4.82 \cdot x_7(v)$

## 6.2. Criar modelos especializados pode ajudar?

Investigamos aqui se treinar modelos diferentes para categorias diferentes de vídeos pode levar erros menores na previsão. A hipótese é que modelos especializados para certas categorias poderiam capturar melhor padrões específicos de cada categoria e ajudar a reduzir os erros. Um potencial problema com essa abordagem é a necessidade de pré-categorizar os vídeos de teste para determinar qual o modelo a ser aplicado. Isso pode ser difícil dependendo do tipo de categorização escolhido. Em todo caso, o objetivo aqui é investigar o *potencial* dessa ideia.

Consideraremos aqui dois tipos de categorização. A primeira é definida diretamente pelas *classes de popularidade* do vídeo (i.e., Memoryless, Viral, etc.). A segunda é definida pelas categorias do sistema do YouTube, que incluem categorias como "Comedy", "Music" e outras. De novo, nossa hipótese é que vídeos em diferentes categorias do YouTube podem possuir padrões de popularidade diferentes, que afetariam a previsão de popularidade. Um modelo mais específico, treinado apenas em vídeo de uma única categoria, poderia capturar essa diferença melhor que um modelo global único.

Analisamos, inicialmente, as classes de popularidade. A Tabela 2 mostra que, para a base de dados *random*, houve uma redução significativa no erro para as classes Junk (cerca de $80\%$) e Quality (quase $63\%$) e reduções estatisticamente insignificantes nas outras duas categorias. Há duas razões principais que explicam esses resultados. Primeiramente, essas são as duas menores classes e a base de dados é muito desbalanceada. Isso significa que o modelo global pode ter sido criado com um *bias* grande em relação a maior classe e, portanto, não generaliza bem para vídeos nas outras classes. Outra razão é que Junk e Quality possuem uma característica muito especial em comum: ambas são caracterizadas por um pico de popularidade súbito e bastante significativo, seguido de uma queda (que é mais rápida para Junk, mas também existe para Quality). Nossa hipótese é que os modelos específicos capturaram melhor essas características que o modelo global.

**Tabela 2. Resultados de mRSE para o modelo ML, comparando os resultados obtidos por um modelo global treinado em vídeos de todas as categorias e um modelo específico para cada categoria ($t_r = 7$, $t_t = 30$).**

| **Vídeos na base de dados *random*** | | | | |
|---|---|---|---|---|
| **Classe** | **# vídeos** | **Modelo Global** | **Modelo Específico** | **Dif. %** |
| Junk | 222 | $0.1402 \pm 0.0274$ | $0.0273 \pm 0.0157$ | -80.5% |
| Memoryless | 3449 | $0.2929 \pm 0.0120$ | $0.2877 \pm 0.0115$ | -1.8% |
| Quality | 948 | $0.1707 \pm 0.0283$ | $0.0644 \pm 0.0098$ | -62.3% |
| Viral | 11504 | $0.1788 \pm 0.0041$ | $0.1738 \pm 0.0043$ | -2.8% |

| **Vídeos na base de dados *top*** | | | | |
|---|---|---|---|---|
| **Classe** | **# vídeos** | **Modelo Global** | **Modelo Específico** | **Dif. %** |
| Junk | 78 | $0.0360 \pm 0.0236$ | $0.0301 \pm 0.0241$ | -16.5% |
| Memoryless | 5129 | $0.1818 \pm 0.0088$ | $0.1811 \pm 0.0087$ | -0.4% |
| Quality | 202 | $0.1229 \pm 0.0310$ | $0.1006 \pm 0.0284$ | -18.2% |
| Viral | 400 | $0.2594 \pm 0.0226$ | $0.1814 \pm 0.0281$ | -30.1% |

Para a base de dados *top*, há algumas diferenças (Tabela 2). Assim como na base *random*, não existe redução significativa em usar um modelo específico para a maior classe (*Memoryless*), provavelmente porque há um *bias* do modelo global para ela (que corresponde a cerca de $88\%$ do total de vídeos). Para vídeos nas classes Junk e Quality, a queda não é estatisticamente significativa. E, por fim, a maior redução de erro ocorre na classe *Viral*. Nessa base de dados, vídeos em todas as categorias tendem a ter picos de popularidade mais distintos [Figueiredo et al. 2011]. As classes diferem, porém, em *quando* esse pico ocorre, como pode ser visto na Tabela 3. Uma fração significativa do total de vídeos possuem esse pico em um dos primeiros 3 dias de vida. Porém, na classe Viral, essa fração é bem menor que nas outras classes. Acreditamos que a grande redução de erro observada no modelo específico para essa classe vem dessa diferença. Isso não ocorre na base de dados *random* porque nela o comportamento de *Memoryless* e *Viral* em relação ao dia de pico é similar.

Esses resultados podem parecer promissores a primeira vista, mas para usá-los na prática é preciso que sejamos capazes de prever corretamente a categoria do vídeo, o que pode ser difícil. Experimentos preliminares em relação a isso mostraram baixa eficácia. Deixamos isso como trabalho futuro.

Há, porém uma outra forma de categorização. Cada vídeo no YouTube é atribuído a uma de uma lista de categorias pré-definidas do sistema (como *News & Politics* ou *Music*). Uma questão deixada em aberto por Szabo e Huberman é se considerar as diferentes

**Tabela 3. Porcentagem de vídeos cujo maior pico de popularidade ocorre no primeiro (ou em um dos 3 primeiros) dias de vida, por classe de popularidade.**

| Classe | Base *random* | | Base *top* | |
|---|---|---|---|---|
| | % Pico no Dia 1 | % Pico Até Dia 3 | % Pico no Dia 1 | % Pico Até Dia 3 |
| Memoryless | 30.1% | 36.5% | 31.5% | 60.5% |
| Viral | 37.8% | 41.6% | 33.8% | 44.9% |
| Junk | 60.1% | 91.5% | 53.9% | 89.7% |
| Quality | 52.9% | 79.6% | 42.7% | 75.5% |

categorias poderia produzir resultados melhores, dado que, intuitivamente, poderíamos esperar que categorias diferentes tivessem padrões de popularidade diferentes.

Criamos, para cada categoria, um modelo específico, treinado e avaliado apenas sobre vídeos daquela categoria. Como pode ser visto na Tabela 4, para a maioria das categorias na base de dados *random*, o modelo específico resulta ou em uma redução estatisticamente insignificante no mRSE ou, em alguns casos, em um *aumento* no erro. Para explicar esses resultados, medimos a distribuição de classes de popularidade por categoria do YouTube nessa base de dados. A maioria das classes possui o mesmo padrão geral: a maior parte dos vídeos é Viral; há uma quantidade considerável, porém bem menor, de vídeos Memoryless; há uma fração pequena de vídeos Quality e uma ainda menor de vídeos Junk. Isso é um indício de que criar modelos específicos por categoria pode não ser útil: não há diferença significativa de comportamento entre categorias. A única exceção onde foi possível obter uma redução de erro maior que $12\%$ nessa base foi a categoria "News & Politics", que é exatamente a que possui a menor porcentagem de vídeos na classe Viral ($58\%$; na base completa a porcentagem é de $70\%$) e a maior porcentagem de vídeos Junk ($3,6\%$, versus $1.4\%$ na base completa). De forma similar, há performance ruim em "Howto & Style", classe que possui uma porcentagem alta de vídeos da classe mais difícil (Memoryless, $26,1\%$) e baixa porcentagem de vídeos da classe mais fácil (Quality, $3,2\%$). Além disso, ela é uma das menores classes nessa base de dados, o que faz com que seja difícil aprender um modelo com bom poder de generalização.

Para a base de dados *top*, há um padrão similar (Tabela 5). Para a maioria das classes, não há ganho estatisticamente significativo em criar um modelo específico. A única exceção é a categoria *Music*, onde há uma redução de $15\%$. Novamente, a maioria das categorias segue o mesmo padrão geral da base em relação à distribuição de classes de popularidade. *Music* se destaca por ter uma proporção muito alta de vídeos *Memoryless* ($96,9\%$, versus $89\%$ na base completa); a outra categoria onde essa proporção é alta é *Comedy*, onde os erros já eram relativamente baixos.

## 7. Conclusões e Trabalhos Futuros

Com a popularização de conteúdo gerado pelos usuários, há uma enorme quantidade de conteúdo na Web, com uma distribuição de popularidade extremamente desigual. Prever a popularidade de conteúdo passa a ser uma tarefa importante para dar suporte a serviços de *cache* ou CDNs, permitindo que administradores desses serviços possam levar em conta a popularidade futura estimada para otimizar o funcionamento destes.

Há diversos padrões de popularidade distintos que vídeos no YouTube podem seguir. Baseado nessa observação, propuzemos um modelo linear simples para prever a popularidade futura baseado em medições periódicas de visualizações no começo da vida

**Tabela 4. mRSE para a base *random* por categoria do YouTube, considerando um modelo global treinado em vídeos de todas as categorias e um modelo específico para cada categoria ($t_t = 30$ dias, $t_r = 7$ dias).**

| Categoria | # vídeos | Modelo Global | Modelo Específico | Dif. % |
|---|---|---|---|---|
| Autos & Vehicles | 561 | $0.1634 \pm 0.0224$ | $0.1684 \pm 0.0222$ | 3.1% |
| Comedy | 609 | $0.2369 \pm 0.0246$ | $0.2404 \pm 0.0253$ | 1.5% |
| Education | 1100 | $0.2141 \pm 0.0173$ | $0.2131 \pm 0.0155$ | -0.5% |
| Entertainment | 2039 | $0.1880 \pm 0.0105$ | $0.1885 \pm 0.0111$ | 0.3% |
| Film & Animation | 902 | $0.2373 \pm 0.0170$ | $0.2254 \pm 0.0191$ | -5.0% |
| Gaming | 239 | $0.1711 \pm 0.0281$ | $0.1765 \pm 0.0310$ | 3.2% |
| Howto & Style | 323 | $0.2508 \pm 0.0396$ | $0.2712 \pm 0.0478$ | 8.1% |
| Music | 3458 | $0.2253 \pm 0.0091$ | $0.2173 \pm 0.0118$ | -3.6% |
| News & Politics | 1352 | $0.1791 \pm 0.0158$ | $0.1562 \pm 0.0135$ | -12.8% |
| Nonprofits & Activism | 360 | $0.1681 \pm 0.0233$ | $0.1698 \pm 0.0229$ | 1.0% |
| People & Blogs | 1443 | $0.1958 \pm 0.0134$ | $0.1952 \pm 0.0127$ | -0.3% |
| Pets & Animals | 272 | $0.2165 \pm 0.0440$ | $0.2174 \pm 0.0361$ | 0.5% |
| Science & Technology | 546 | $0.1979 \pm 0.0206$ | $0.1989 \pm 0.0201$ | 0.5% |
| Sports | 1797 | $0.1805 \pm 0.0150$ | $0.1807 \pm 0.0160$ | 0.1% |
| Travel & Events | 1117 | $0.1904 \pm 0.0143$ | $0.1914 \pm 0.0150$ | 0.5% |

**Tabela 5. mRSE para a base *top* por categoria do YouTube, considerando um modelo global treinado em vídeos de todas as categorias e um modelo específico para cada categoria ($t_t = 30$ dias, $t_r = 7$ dias).**

| Categoria | # vídeos | Modelo Global | Modelo Específico | Dif. % |
|---|---|---|---|---|
| Autos & Vehicles | 386 | $0.1982 \pm 0.0487$ | $0.1892 \pm 0.0354$ | -4.5% |
| Comedy | 321 | $0.1310 \pm 0.0251$ | $0.1394 \pm 0.0358$ | 6.4% |
| Education | 431 | $0.2484 \pm 0.0317$ | $0.2501 \pm 0.0333$ | 0.7% |
| Entertainment | 368 | $0.1575 \pm 0.0256$ | $0.1565 \pm 0.0261$ | -0.6% |
| Film & Animation | 325 | $0.2287 \pm 0.0346$ | $0.2276 \pm 0.0361$ | -0.5% |
| Gaming | 467 | $0.1133 \pm 0.0202$ | $0.1220 \pm 0.0260$ | 7.6% |
| Howto & Style | 458 | $0.1764 \pm 0.0253$ | $0.1745 \pm 0.0256$ | -1.1% |
| Music | 334 | $0.2762 \pm 0.0326$ | $0.2348 \pm 0.0348$ | -15.0% |
| News & Politics | 423 | $0.1391 \pm 0.0252$ | $0.1332 \pm 0.0247$ | -4.3% |
| Nonprofits & Activism | 467 | $0.1823 \pm 0.0317$ | $0.1871 \pm 0.0330$ | 2.7% |
| People & Blogs | 409 | $0.1666 \pm 0.0288$ | $0.1665 \pm 0.0289$ | -0.1% |
| Pets & Animals | 314 | $0.1828 \pm 0.0314$ | $0.1873 \pm 0.0338$ | 2.5% |
| Science & Technology | 424 | $0.1795 \pm 0.0277$ | $0.1810 \pm 0.0287$ | 0.8% |
| Sports | 297 | $0.1845 \pm 0.0426$ | $0.2055 \pm 0.0620$ | 11.4% |
| Travel & Events | 381 | $0.2091 \pm 0.0315$ | $0.2123 \pm 0.0318$ | 1.5% |

de um vídeo. Nosso modelo obteve reduções significativas no erro relativo quadrático médio em relação a um modelo previamente proposto que considera apenas o total agregado de visualizações e, portanto, não consegue distinguir entre vídeos com padrões de popularidade diferentes.

Exploramos também a questão de se é interessante produzir modelos especializados para tipos diferentes de vídeos. Notamos que, se for possível prever qual é o padrão de popularidade seguido pelo vídeo, reduções significativas no erro de previsão podem ser obtidas. Porém, identificar o padrão geral de popularidade com base apenas no padrão de acesso inicial não é um problema trivial, e deixamos isso como trabalho futuro. Já criar modelos específicos para cada categoria de vídeo do YouTube não trouxe ganhos exceto

em raros casos de categorias isoladas.

Há uma série de outras fontes de informações que podem ser úteis para previsão de popularidade, tal como o número de comentários recebidos pelo vídeo, e mesmo informações sobre o usuário que o postou (como a popularidade média dos demais vídeos dele). Experimentos preliminares não trouxeram ganhos, porém, porque a maioria dessas informações é altamente correlacionada com o número de visualizações. Acreditamos que modelos não-lineares podem ser capazes de aproveitar essas informações e gerar resultados de previsão melhores, e é essa a principal direção futura de trabalho.

## Agradecimentos

Esta pesquisa é parcialmente financiada pelo Instituto Nacional de Ciência e Tecnologia para a Web - INCTWeb (MCT/CNPq 573871/2008-6), CNPq, CAPES e FAPEMIG.

## Referências

Alexa.com (2011). Alexa top 500 global sites. `http://www.alexa.com/topsites`. [Online; acessed 2-November-2011].

Cha, M., Kwak, H., Rodriguez, P., Ahn, Y., and Moon, S. (2007). I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system. In *Proc. of the 7th ACM SIGCOMM conf. on Internet measurement*. ACM.

Cha, M., Kwak, H., Rodriguez, P., Ahn, Y., and Moon, S. (2009). Analyzing the video popularity characteristics of large-scale user generated content systems. *IEEE/ACM Transactions on Networking (TON)*, 17(5).

Crane, R. and Sornette, D. (2008). Robust dynamic classes revealed by measuring the response function of a social system. *Proc. of the National Academy of Sciences*, 105(41).

Figueiredo, F., Benevenuto, F., and Almeida, J. (2011). The tube over time: Characterizing popularity growth of youtube videos. In *Proc. of the 4th ACM Intl. Conf. of Web Search and Data Mining (WSDM'11)*.

Gonçalves, M., Almeida, J., Santos, L., Laender, A., and Almeida, V. (2010). On popularity in the blogosphere. *Internet Computing*.

Heffernan, V. (2009). Uploading the avant-garde. http://www.nytimes.com/2009/09/06/magazine/06FOB-medium-t.html. [Online; acessed 2-November-2011].

Lee, J., Moon, S., and Salamatian, K. (2010). An approach to model and predict the popularity of online contents with explanatory factors. In *Intl. Conf. on Web Intelligence and Intelligent Agent Technology*. IEEE.

Lerman, K. and Hogg, T. (2010). Using a model of social dynamics to predict popularity of news. In *Proc. of the 19th Intl. Conf. on WWW*. ACM.

Liu, D. and Nocedal, J. (1989). On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1).

Rodrigues, T., Benevenuto, F., Almeida, V., Almeida, J., and Gonçalves, M. (2010). Equal but different: A contextual analysis of duplicated videos on youtube. *Springer Journal of the Brazilian Computer Society*, 16(3).

Szabo, G. and Huberman, B. (2010). Predicting the popularity of online content. *Communications of the ACM*, 53(8).

Wu, F. and Huberman, B. (2007). Novelty and collective attention. *Proc. of the National Academy of Sciences*, 104(45).

Wu, T., Timmers, M., De Vleeschauwer, D., and Van Leekwijck, W. (2010). On the use of reservoir computing in popularity prediction. In *2010 2nd Intl. Conf. on Evolving Internet*. IEEE.