Predict Topic Trend in Blogosphere

Jack Guo 05596882 jackguo@stanford.edu

Abstract

Graphical relationship among web pages has been used to rank their relative importance. In this paper, we introduce a way to leverage graphical relationship to capture characteristics of information diffusion. With this approach, we extract subgraphs by projecting the first few domains mentioning a topic onto the domain hyperlink graph and co-mention graph, and use machine learning method to construct a predictive model. Then we use a greedy feature selection algorithm to extract the most helpful features. Simulation result shows we are able to classify topic popularity with 78% accuracy by observing even only first 5 or 10 mentioning domains.

Introduction

Blogosphere serves as an important medium for information propagation and diffusion. Leskovec et al.[3] developed the meme-tracker application to track memes in the blogosphere, which enable us to identify the trend and cycle of a meme over time. Some memes capture public attention and propagate broadly and persistently through the blogosphere, while some memes fade away quickly after attracting attention of a few blogs. This paper is trying to answer the following questions: Given the first few mentioning URLs of a meme, how can we predict whether this meme will be popular? Given a pair of meme with similar starting trend, how can we predict which meme will be more popular than the other? What are the characteristics of popular and unpopular memes, and which features are most helpful?

Graphical features are useful in characterizing the underlying interaction in the network and thus help in making predictions. Leskovec et al.[1] introduced a graphical based approach to learn from contextual subgraphs of the webpages to predict relative quality of the search results. And Shi et al.[2] demonstrates the effectiveness of exploiting network features in characterizing high/medium/low citation papers. This paper trains a predictive model using a rich set of contextual graphical features and utilizes forward greedy feature selection algorithm to identify most helpful features.

The dataset we are using is Spinn3r data from June to Aug 2011, which indexes the blogosphere at a rate of 1 million posts per hour. The dataset has a total compressed file size of around 500GB with around 2 billion posts.

Problem Description

We formally define the classification and differentiation tasks of the meme as follows.

Classification task: Given the first N mentioning URLs of the meme, predict whether this meme will become popular in the future. Memes are labeled as high or low popularity if it is above a given high popularity threshold (say 200 mentions) or below a given popularity threshold (say 100 mentions), respectively.

Differentiation task: For a pair of meme with similar starting trend (say 5 hours), predict which meme will be more popular than the other.

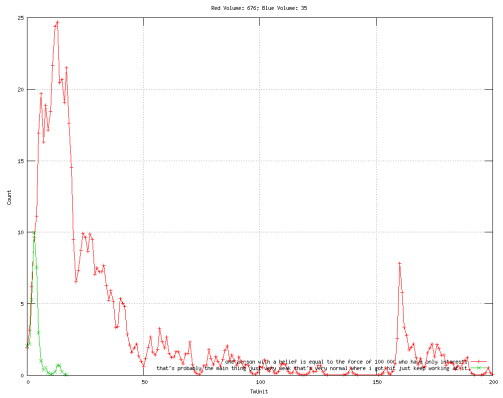


Figure 1: Two memes with similar beginning phase. One meme becomes popular while the other fades soon. **Red Meme**: “One person with a belief is equal to the force of 100000 who have only interests”, the paraphrased John Stuart Mill quote that the Norwegian terrorist Anders Behring Breivik tweeted before his murderous rampage. **Green Meme**: "That's probably the main thing. Just very weak. That's very normal where I got hit. Just keep working at it.", a quote from White Sox 1B Paul Konerko when he is out of lineup.

Meme Extraction and Clustering

We extract quotes in order to track memes, because quotes subject to only small changes through the propagation process. Clustering different quote variants of a meme enables us to identify the time profile of a meme. Leskovec et al.[3] provides a method to create links between quote variants if they are textually similar and then use a greedy approach to partition the resulting DAG graph in order to retrieve quote clusters. Although this approach has good accuracy (~100%) but suffers from a low recall (~50%) because many irrelevant but textually similar quote variants are classified in the same cluster. Here we propose a better clustering algorithm.

Firstly we hash all 4-word shingles of a quote into hash table, because if two quote variants are from the same cluster, it is almost for sure they will have a 4-word overlap, so they will have a chance to matchup in a hash table bucket. Secondly we evaluate the “quality” of each hash bucket by computing the average pairwise quote approximate edit distance and discard bad

buckets. Then we perform a pairwise comparison within each bucket to identify textual similar quote and link them in the quote graph. Finally we use quote timestamp information to delete links that are temporally far apart and extract the connected components as quote cluster.

This method has high accuracy (~100%) and high recall (>95%). The algorithm runs for less than 30 minutes for clustering 4 million quote variants, which is 100 times faster than previous algorithm. This clustering algorithm enables us to handle large dataset and extracts meme precisely, which is critical to the trend prediction task.

Feature Extraction

We identify 4 types of subgraphs and extract 64 graphical features and 4 quote features. Graphical features include node number, edge number, triad number, maximum degrees, number of components, size of gcc, gcc node number, gcc edge number, density, clustering coefficient, excess degree entropy, in/out-degree for each type of subgraph. 4 quote features are number of words, number of characters in the quote, and media domain ratio, average domain mentioning time (some domains tend to mention memes earlier than others). Next we will describe notion and definition of the several types of networks.

- Domain Hyperlink Graph is defined as the graph with all domains as nodes and hyperlinks between posts of the domains as directed edges.
- Domain Co-mention Graph is defined as the graph with all domains as nodes and the number of co-mentioning memes between two domains as the weight of their undirected edge.
- Hyperlink Projection Graph (PROJ) is a subgraph of the Domain Hyperlink Graph induced on N domains observed. Hyperlinks between domains are indicative of meme diffusion.
- Hyperlink Connection Graph (CONN) is a subgraph of the Domain Hyperlink Graph induced on N domains together with connector domains which connect the N domain together into one weakly connected component. The role of hyperlink connection graph is similar to that of hyperlink projection graph.
- Co-mention Projection Graph (COM) is a subgraph of the Domain Co-mention Graph induced on N domains observed. Co-mention relationship represents the common topics and interests between two domains. If the co-mention edge weight is high between two domains, then they have similar tastes or belong to the same community.
- Weighted Hyperlink Projection Graph (MIX) is a weighted subgraph of the Domain Hyperlink Graph with hyperlink as the direction of the directed edge and percentage of co-mentions as edge weight.
- Non-network features (NONNET) are general features related to the meme or with the mentioning domains such as number of the words of the meme quote, or percentage of media domains mentioning the meme.

- Time feature (TIME) is a single feature defined as the time span of the first several mentions observed. Intuitively it is most indicative of meme popularity, because given the number of domain observed N as fixed, if a meme reaches N mentions in shortest time as possible, then it is more likely to be popular in the future. We use this feature as a baseline to see if our model using graphical features efficiently capture the time dynamics of the meme diffusion.

For graphical features, we use usual graphical features such as node number, edge number, triad number, component number, maximum degrees, density, clustering coefficient, average in/out-degrees. And we also use the excess degree entropy feature which is defined as[2]:

$$q(k) = \frac{(k+1)P_{k+1}}{< k >}$$

$$H(q) = -\sum_{k=1}^{N} q(k)\log(q(k))$$

where P_k is the number of nodes with degree k. <k> is the average degree.

Another set of features we used are features in the fringe of the projection graph. The fringe of projection graph is defined as the outside nodes connected by nodes inside projection graph. We count the number of nodes in the fringe of the projection and edges between them.

For the weighted hyperlink projection graph, we define subscriber (in-link) competition coefficient and source (out-link) competition coefficient features as average Jaccard similarity between their subscriber (in-link) set and source (out-link) set, respectively. The figure below illustrates the examples of high and low subscriber competition coefficient between two nodes. The competition coefficient shows the competition caused in meme diffusion, and it also reflects the similarity between two domains, i.e. if their subscriber/sources are very similar, then the two domains are very similar as well.

(a) (b)

Figure 2**:** Illustration of (a) high subscriber competition coefficient between two nodes and (b) low subscriber competition coefficient between two nodes. Two nodes in (a) has subscriber competition coefficient of 3/5 and in (b) has that of 1/7.

For a complete set of features used in the model with detailed description, please see the Appendix.

Learning Model

We adopt a logistic regression model for both tasks. For the classification task, we use a linear model that combines the features of the meme; and for the differentiation task, we take features as the difference between the feature vectors of the pair of meme in consideration. For both models, we use a 5-fold cross validation to compute the prediction accuracy.

An alternative model for this problem is to use domain vector as features, each domain vector entry being 1 if a domain mentions the meme in the beginning phase considered, and 0 otherwise. We could build a logistic regression with domain vector feature to train a predictive model. This model will serve as a baseline model (denoted as Domain Regression below). Training and test data are prepared so that there are equal number of high and low popularity memes so that a random baseline model will give a prediction accuracy of 50%.

Classification Task Results

The classification results for different feature combination and different high/low popularity meme threshold are shown in table 1. Result of domain regression is also shown as a comparison. From the result we can see that co-mention graph features and weighted hyperlink graph features are especially helpful and we could achieve around 80% classification accuracy using graphical feature model and around 70% using domain regression.

From the table 1 we reach the following conclusions: (1) We can achieve around 80% of prediction accuracy using graphical features, which is about 10% better than the domain feature baseline model. (2) We can achieve around 80% of prediction accuracy in the classification task. Although time feature has very good prediction accuracy, it doesn't help much if we use full contextual graphical features. This means the graphical features can effectively capture the time evolution of the meme popularity growth. (3) Using quote cluster meme gives us better results because quote cluster integrates mentions from its quote variants and thus better characterizes the meme trend.

Table 2 shows the prediction accuracy of logistic regression model using single feature. From this table we can identify the features that are most relevant to the meme popularity. In the next section, we will use greedy forward feature selection to pick the set of features that best capture the trend of the meme.

Table 1: Classification task results for different feature sets, and different meme formulation. The high and low popularity thresholds are 240 and 80, respectively. The number of mentions observed is 30. Each result is averaged over 1000 runs. “Variant” means quote variant meme and “Cluster” means quote cluster meme.

Feature Set (# of features)	**Variant**	**Cluster**
PROJ (10)	59.6%	63.1%
NONNET (4)	58.0%	56.9%
COM (12)	72.5%	74.1%
CONN (5)	58.5%	60.7%
MIX (12)	75.7%	76.2%
TIME (1)	69.4%	71.3%
PROJ+COM (22)	74.6%	78.7%
PROJ+CONN (15)	59.9%	64.6%
PROJ+COM+CONN (27)	74.0%	78.6%
COM+CONN+MIX (29)	75.7%	79.6%
PROJ+COM+MIX (29)	75.9%	80.3%
PROJ+COM+CONN+MIX (34)	75.4%	80.6%
ALL (38)	**76.5%**	**81.1%**
ALL+TIME (39)	**77.0%**	**82.9%**
Domain Regression	66.1%	72.3%

Table 2**:** Prediction accuracy of the classification task using only each single feature. (quote cluster meme)

PROJ Graph		**NONNET**		**COM Graph**	
ProjTriadNum	**61.1%**	NonNetMBRatio	58.0%	CoMEdgeNum	51.6%
ProjCompNum	58.3%	NonNetReportTime	51.7%	CoMTriadNum	52.7%
ProjMaxDeg	60.1%	NonNetWordNum	50.6%	CoMCompNum	50.3%
ProjDensity	60.9%	NonNetCharNum	50.4%	CoMGccEdgeNum	50.4%
ProjClusterCoeff	60.8%			CoMMaxDeg	47.7%
ProjEntExcessDeg	**61.8%**			**CoMDensity**	**71.7%**
ExtLinkNodeNum	60.6%			**CoMClusterCoeff**	**59.0%**
ExtLinkEdgeNum	59.9%			CoMEntExcessDeg	49.8%
DegLinkIn	57.4%	**MIX Graph**		ExtCoMNodeNum	59.9%
DegLinkOut	55.0%	MixDensity	56.1%	ExtCoMEdgeNum	59.7%
		MixClusterCoeff	59.8%	DegCoMIn	58.1%
CONN Graph		**MixInCompCoeff**	**69.0%**	DegCoMOut	48.0%
ConnNodeNum	50.2%	**MixOutCompCoeff**	**66.1%**		
ConnEdgeNum	61.7%	**DegMixIn**	**62.1%**		
ConnTriadNum	61.9%	DegMixOut	60.1%	**TIME**	
ConnDensity	56.1%	DegMixExtIn	61.1%	**TimeSpan**	**71.3%**
ConnClusterCoeff	54.2%				

Greedy Model Feature Selection

We use the forward greedy model feature selection to identify key features/characteristics that differentiates high/low popularity meme. In each step, we add one feature to the feature set so that it achieves best prediction accuracy.

Table 3: Greedy forward feature selection (quote cluster meme). The time feature is excluded in the selection process.

Step	Feature selected in each step	Graph Type	Prediction Accuracy
1	**CoMDensity**	COM	69.8%
2	**DegMixIn**	MIX	76.0%
3	CoMClusterCoeff	COM	77.2%
4	**MixInCompCoeff**	MIX	77.3%
5	NonNetMBRatio	NONNET	78.5%
6	**ExtLinkEdgeNum**	PROJ	79.4%
7	DegCoMOut	COM	79.9%
8	ExtCoMEdgeNum	COM	80.7%
18	…	…	82.3%

Table 4**:** Greedy forward feature selection (quote variant meme). The time feature is excluded in the selection process.

Step	Feature selected in each step	Graph Type	Prediction Accuracy
1	**CoMDensity**	COM	68.9%
2	**DegMixIn**	MIX	73.9%
3	NonNetWordNum	NONNET	75.5%
4	ConnClusterCoeff	CONN	76.4%
5	**MixInCompCoeff**	MIX	77.9%
6	**ExtLinkEdgeNum**	PROJ	78.5%
7	NonNetReportTime	NONNET	78.8%

The sets of features selected for the two settings (cluster/variant as meme) are quite similar. CoMDensity and DegMixIn are the two most important features and they combined achieved 76% and 73.9% prediction accuracy, only a 5~6% degradation from the full model with all 38 features. Moreover, MixInCompCoeff and ExtLinkEdgeNum are also common top picks for both cases. The consistency between two feature sets shows that the models for the two cases are consistent.

From the greedy feature selection process above, we can see that high popularity meme has low co-mention density (CoMDensity) and clustering coefficient (CoMClusterCoeff), meaning the domains participate in mentioning the meme have different interests. Intuitively this is true because it allows the meme to propagate in different parts of the topic/interest domains. High popularity meme also has high number of subscribers (DegMixIn) and low subscriber competition coefficient (MixInCompCoeff), because more subscribers and low competition coefficient means more influence and a larger body of target meme diffusion nodes. Moreover,

high popularity meme has high connection clustering coefficient (ConnClusterCoeff) and high number of edges between nodes linked to the projection graph from outside (ExtLinkEdgeNum) because closed hyperlink triangle implies more reliable meme diffusion channel.

The figure below is an exemplary illustration of the projection graphs of the high/low popularity memes with beginning 5 mentioning domain observed. The directed edges are hyperlinks and the distance between nodes represents their co-mention relationship: the closer the two nodes are, the higher the co-mention edge weight is. In (a) the nodes are far apart from each other and have each a rich community of subscribers, so it has low co-mention density and clustering coefficient, high number of subscribers and low subscriber competition coefficient, while in (b) the nodes are close to each other meaning they have similar interest, and the meme diffusion is restricted to a small region.

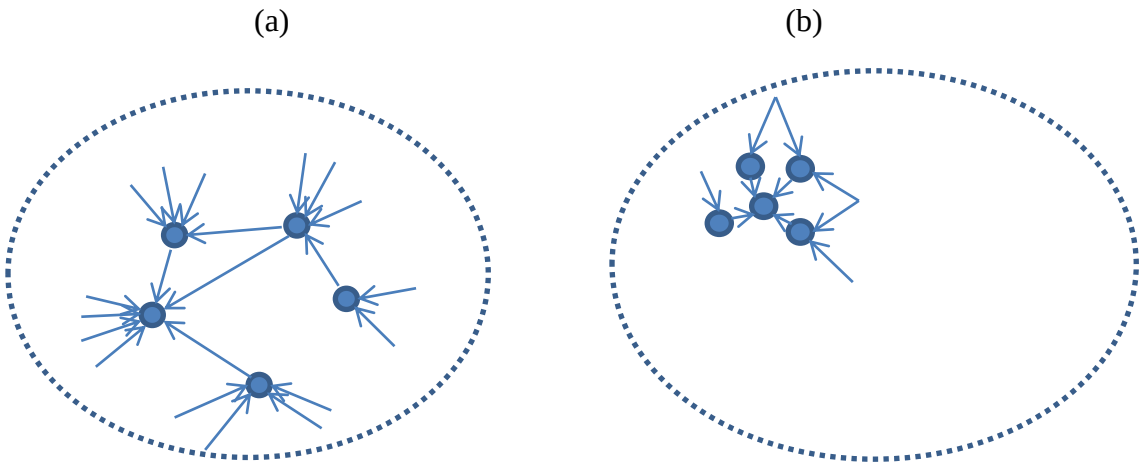


Figure 3**:** Example of projection graph of (a) high and (b) low popularity meme with 5 domains. Dash line indicates the boundary of blogosphere. Distance between nodes represents the co-mention relationship. The closer the nodes are, the more weight they have in the co-mention graph.

Sensitivity analysis of Number of Domains Observed

In this section, we investigate how the classification accuracy varies as the number of domains/URLs observed in the beginning phase of a meme changes (from 5 to 60).

From Table 5 we could see that we can make good classification by only looking at the first 5~10 domains; the prediction accuracy first increase with number of domains, but then falls off as the number of domains observed goes beyond 30; logistic regression method using graphical features is much better compared to Domain SVM especially when the number of domains observed is too few.

Table 5**:** Classification accuracy with different number of observed domains (Quote cluster meme). Features are calculated using the first 5, 10, 20, 30, 40, 50, 60 domains.

# of domain observed	**5**	**10**	**20**	**30**	**40**	**50**	**60**
PROJ (10)	59.8%	62.0%	65.0%	63.1%	63.8%	63.2%	64.5%
NONNET (4)	61.8%	62.9%	58.1%	58.7%	61.6%	61.9%	65.0%
COM (12)	76.1%	75.8%	75.1%	74.6%	73.8%	72.7%	73.5%
CONN (5)	57.0%	57.9%	62.0%	61.7%	62.5%	62.3%	62.1%
MIX (12)	72.5%	73.7%	76.0%	76.5%	76.5%	75.9%	75.4%
PROJ+COM (22)	77.0%	77.2%	77.7%	77.7%	76.8%	75.9%	76.4%
PROJ+CONN (15)	62.0%	64.1%	66.6%	64.5%	64.8%	64.7%	65.6%
PROJ+COM+CONN (27)	76.9%	77.4%	78.2%	78.5%	77.9%	76.4%	77.1%
COM+CONN+MIX (29)	76.9%	77.3%	78.7%	79.3%	78.5%	76.7%	76.6%
PROJ+COM+MIX (29)	77.2%	78.0%	79.1%	80.3%	79.3%	77.4%	77.7
PROJ+COM+CONN+MIX (34)	77.2%	78.2%	79.0%	80.6%	79.5%	77.3%	77.3%
ALL (38)	**77.5%**	**78.1%**	**80.5%**	**81.5%**	**81.2%**	**80.2%**	**80.6%**
Domain Regression	64.7%	66.8%	70.3%	72.2%	70.3%	70.6%	72.3%

Table 6**:** Prediction accuracy of differentiation task. We pick meme pairs that are similar in the beginning 3 hours and 5 hours.

Feature Set (# of features)	**Variant-5h**	**Variant-3h**	**Cluster-5h**	**Cluster-3h**
PROJ (10)	84.5%	83.0%	81.9%	82.1%
NONNET (4)	70.7%	66.1%	71.7%	62.8%
COM (12)	**81.3%**	**87.4%**	**86.9%**	**86.3%**
CONN (5)	70.5%	73.6%	75.6%	76.3%
MIX (12)	**85.8%**	**87.4%**	**86.9%**	**85.4%**
TIME (1)	45.1%	56.9%	55.3%	54.0%
PROJ+COM (22)	85.3%	85.8%	85.4%	87.5%
PROJ+CONN (15)	83.9%	85.1%	82.3%	83.5%
PROJ+COM+CONN (27)	83.2%	86.3%	86.8%	88.1%
COM+CONN+MIX (29)	84.5%	86.1%	87.8%	88.1%
PROJ+COM+MIX (29)	85.3%	85.3%	87.3%	87.0%
PROJ+COM+CONN+MIX (34)	83.8%	87.1%	87.6%	89.1%
ALL (38)	**83.7%**	**86.7%**	**88.3%**	**88.7%**
ALL+TIME (39)	84.9%	87.9%	88.5%	89.1%

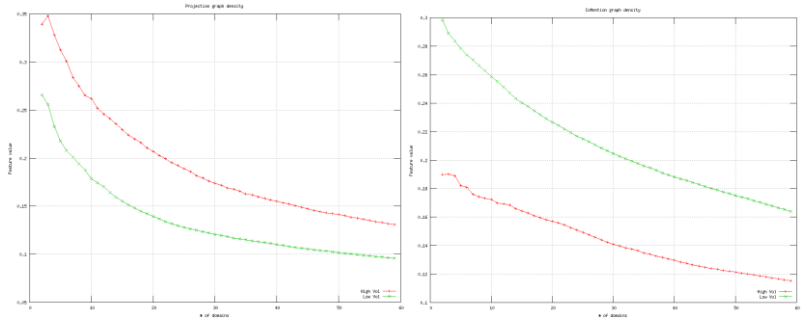
Differentiation Task Results

Table 6 shows the differentiation task results for quote variant meme and quote cluster meme with first 3 or 5 hours of mentions observed. We can achieve around 88% of prediction accuracy for quote cluster meme. That means given two memes that are indistinguishable in trends for the first 3 or 5 hours, we can reliably predict (with >88% accuracy) which meme will be more popular than the other.

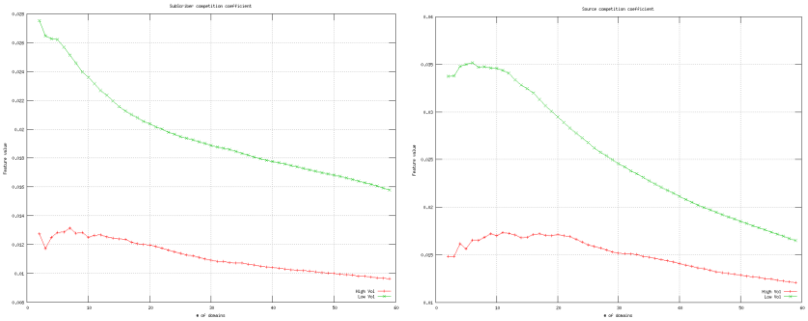
Feature Evolution

In this section, we will study the evolution of feature values as more and more mentioning domains are observed. For each given number of observed domains, we compute the average feature values of high and low popularity memes, and plot them in Fig 4. The horizontal axis denotes the number of domain observed and the vertical axis denotes the average feature values. Red curve is for high popularity memes and green curve is for low popularity memes. We can see from the figures that the feature values of low popularity meme decreases twice as fast as that of high popularity meme for co-mention graph density, subscriber and source competition coefficients features. This is due to the fact that for low popularity meme, the meme propagates to the end consumer domains very quickly, and end consumer domains (small blog domains) have low co-mention relationship and low subscriber competition coefficient and low source competition coefficient with other average domains.

(a) Projection Graph Density **(b) Co-Mention Graph Density**



(c) Subscriber Competition Coefficient **(d) Source Competition Coefficient**



(e) Average Weighted Graph In-Degrees

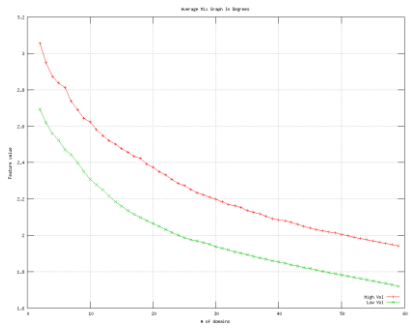


Figure 4: Evolution of average feature values of high popularity meme and low popularity meme (quote cluster meme). Features include projection graph density, co-mention graph density, subscriber/source competition coefficients and average weighted graph in-degrees.

Role of Media and Blog Domains

In this section, we investigate the roles media and blog domains play in meme diffusion. Specifically, we manually labeled a few hundreds of media domains in the data, such as nytimes.com, cnn.com etc, and labeled the rest as blog domains. Then we make prediction using only information of one type of domains.

Table 7: Predict meme popularity by using set of features calculated with media/blog domains only. "-M" and "-B" suffix denotes observing media domain and blog domain only, respectively.

Method (# of features)	Variant-M	Variant-B	Cluster-M	Cluster-B
PROJ (10)	60.9%	61.7%	65.0%	63.5%
NONNET (4)	59.1%	57.4%	52.6%	49.7%
COM (12)	70.6%	65.8%	72.2%	68.2%
CONN (5)	54.7%	55.5%	60.8%	58.3%
MIX (12)	75.2%	65.5%	75.4%	67.6%
TIME (1)	65.8%	46.7%	60.3%	47.6%
PROJ+COM (22)	73.3%	67.5%	76.2%	71.3%
PROJ+CONN (15)	62.1%	60.7%	65.8%	63.1%
PROJ+COM+CONN (27)	72.8%	67.0%	76.2%	71.3%
COM+CONN+MIX (29)	74.4%	67.6%	76.5%	72.0%
PROJ+COM+MIX (29)	74.6%	68.5%	76.9%	72.3%
PROJ+COM+CONN+MIX (34)	74.0%	67.9%	77.1%	72.3%
ALL (38)	**75.8%**	**69.5%**	**77.8%**	**72.5%**
ALL+TIME (39)	76.4%	69.4%	77.7%	72.4%

We can see that if we observe only media domains, the prediction accuracy is comparable to that of using both types of domains. So media domains serve as the backbone of meme diffusion and we could infer the meme behavior fairly well from media domains; but for observing only blog

domains, the results are poorer because generally blog domains serve as end consumer of the meme so it doesn't give us as much clue of meme diffusion media domains.

Conclusion and Future Work

In this paper we define the classification and differentiation tasks of memes based on its popularity and construct predictive models based on contextual graphical features. We demonstrate that with graphical features we can achieve around 80% classification accuracy for classification tasks and we could make good prediction even if we can only observe first 5 to 10 domains/URLs that mentioning a meme; and we can achieve around 88% prediction accuracy for the differentiation task. These results demonstrate the effectiveness of contextual graphical features.

For future work, we can try to construct the theoretical model to capture information diffusion process in blogosphere.

References

[1] J. Leskovec, S. Dumais, E. Horvitz. *Web Projections: Learning from Contextual Subgraphs of the Web.* International World Wide Web Conference (WWW), 2007.

[2] X. Shi, J. Leskovec, D. A. McFarland. *Citing for High Impact* Joint Conference on Digital Libraries (JCDL), 2010.

[3] J. Leskovec, L. Backstrom, J. Kleinberg. *Meme-tracking and the Dynamics of the News Cycle* ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009.

Appendix

Graph	Abbreviation	Description
Hyperlink Projection Graph (PROJ) 10 features	ProjTriadNum	Number of triads. Measure the number of "closed triangles" in meme diffusion through links.
	ProjCompNum	Number of components
	ProjMaxDeg	Maximum degrees
	ProjDensity	Edge density
	ProjClusterCoeff	Clustering coefficient
	ProjEntExcessDeg	Excess degree distribution entropy
	ExtLinkNodeNum	Number of nodes linked from outside of the projection graph
	ExtLinkEdgeNum	Number of edges between nodes linked from outside of the projection graph
	DegLinkIn	Average in-degrees of the nodes. Measure potential meme diffusion target size
	DegLinkOut	Average out-degrees of the nodes. Measure susceptibility of the nodes in meme diffusion
Hyperlink Connection Graph (CONN) 5 features	ConnNodeNum	Number of nodes
	ConnEdgeNum	Number of edges
	ConnTriadNum	Number of triads
	ConnDensity	Edge density
	ConnClusterCoeff	Clustering coefficient
Co-mention Projection Graph (COM) 12 features	CoMEdgeNum	Number of edges
	CoMTriadNum	Number of triads
	CoMCompNum	Number of components
	CoMGccEdgeNum	Number of edges in the largest connected components
	CoMMaxDeg	Maximum degree
	CoMDensity	Edge density
	CoMClusterCoeff	Clustering coefficient
	CoMEntExcessDeg	Excess degree distribution entropy
	ExtCoMNodeNum	Number of nodes linked from outside of the co-mention graph
	ExtCoMEdgeNum	Number of edges between nodes linked from outside of the co-mention graph
	DegCoMIn	Average in-degrees of the nodes
	DegCoMOut	Average out-degrees of the nodes
Weighted hyperlink Projection Graph (MIX) 7 features	MixDensity	Edge density
	MixClusterCoeff	Clustering coefficient
	MixInCompCoeff	Subscriber (in-link) competition coefficient, defined as average Jaccard similarity between sets of subscribers (in-links) of two nodes
	MixOutCompCoeff	Source (out-link) competition coefficient, defined as average Jaccard similarity between

		sets of sources (out-links) of two nodes
	DegMixIn	Average in-degrees of the nodes
	DegMixOut	Average out-degrees of the nodes
	DegMixExtIn	Average in-degrees of the nodes linked from outside of the projection graph
Non-network Features (NONNET) 4 features	NonNetMBRatio	Percentage of media domains in the first several mentioning domains observed
	NonNetReportTime	Average domain report time. Some domains tend to report meme earlier than others. Average report time of each domain is calculated as the average lag with respect to median mention time
	NonNetWordNum	Number of words in the quote that represents the meme
	NonNetCharNum	Number of characters in the quote that represents the meme
Time Feature (TIME) 1 feature	TimeSpan	Time span of the first several mentions observed.

Appendix Table 1: Summary of features used.