# A Model of Random Walks on Bipartite Graph for Classification Problems


## Abstract

We introduce a general framework of semi-supervised graphical model that is applicable to any classification tasks that can be represented with norminal features, requiring only a handhul of labeled examples. We conducted experiments on three different NLP tasks: prepositional phrase attachment, named entity classification, and domain-specific terminology extraction. We show both theoretically and empirically how the graph structure helps making use of large amount of unlabeled data. Based on the empirically study of underlying correlations between graph structures and classification performance, we incorporate active learning techniques to achieve the best learning rate with minimum requirement on labeled data.


## 1 Introduction

Semi-supervised learning on graphs is a direction that has drawn great attentions from NLP researchers. An abstract framework that we have often seen in graph based NLP is constructing a graph in which each vertex represents an instance which can be a word, a concept, a sentence, an article etc. Edges are connected and weighted according to some manually selected function that defines the closeness or similarity of any two instances in the context of the application. One possible usage of the graph is to predict the property (label) of a node via comparing the distances of this node to other nodes with known labels. For example, //add a citation here

Another information we can obtain from the graph is the importance of a node. We see algorithms such as PageRank(), HITS(), and graph based summarization systems() .

In this paper, we introduce a model that shares with many other graphical methods the idea of encoding similarities between instances into a graphical structure and learning from unlabeled instances, while presents its novelty in the following two aspects. First, the graph is bipartite. Example nodes and feature nodes form two subsets of the bipartite graph. We will show later this can be equivalently converted to a graph consisting of only example nodes assuming a special edge weight definition. This assumption simplifies the graph construction by a factor of the number of example nodes. Second, we observe from experiment that when the labeled training size is very small compared to the entire graph, performance of the model is unstable due to the randomness in sampling training examples. We extend the model with an active learning technique that intellegently chooses the most "informative" unlabeled examples to learn. Such property is well appreciated in applications where labeled examples are very expensive to obtain.

## 2 The Bipartite Graph Model

### 2.1 Graph Construction

We first formulate a classification task with the following parameters:

a. A set of $n$ examples, among which only $p$ of them has been labeled ($n >> p$). The $p$ labeled examples and their labels form a training set: $(X_1, y_1), (X_2, y_2), ..., (X_p, y_p)$;

b. Denote the rest of $n - p$ unlabeled examples $X_{p+1}, X_{p+2}, ...X_n$;

c. Each example's feature is a subset of the norminal feature set $F = \{f_1, f_2, ..., f_m\}$. Write $X_i = (X_i^1, X_i^2, ..., X_i^m)$, and $X_i^j = 1$ if example $X_i$ has nominal feature $f_j$, otherwise $X_i^j = 0$.

The goal is to predict the labels of any (all) unlabeled examples.

To construct the bipartite graph $G = (X, F, E)$ ($X, F$ are node sets and $E$ is edge set), we

create examples nodes for each example $X = \{X_1, X_2, ..., X_n\}$, and feature nodes for each nominal feature $F = \{f_1, f_2, ..., f_m\}$. We connect an example node and a feature node using an undirected edge if and only if the feature is present in the example, $(X_i, f_i) \in E$ iff $X_i^j = 1$.

### 2.2 Random Walk

// Describe the random walk hitting time method.

// add algorithm

### 2.3 Label Propagation

An alternative impelmentation that solves the same problems as performing random walks is label propagation.

// add algorithm of tumbl

// explain or prove the equivalence of the two algorithms theoretically

// cite Zhu's 2002 paper

## 3 Active Learning

Observations in a pilot experiment on the preporsitional phrase attachment dataset motivated applying active learning. In this section, we first briefly describe the PP attachment classification task, dataset, and results form the pilot experiment. Then we present the active learning algorithm.

### 3.1 PP Attachment Dataset

1) ppattach disambiguation

2) dataset

3) state-of-art, backoff method

### 3.2 Pilot Experiment

use basic tumbl model

plot: learning curve from 10 to 1000 training size

plot: uncertianty - when sample size is small (e.g. 10 , 50) performance

variance is large, motivate a method to better choose examples to be labeled

### 3.3 Algorithm with Active Learning

// need to fill in after experiments

## 4 Experiments

### 4.1 PP attachemnt dataset

1) baseline: backoff method

2) tumbl, randomly pick labeled examples

3) active learning

Todo: describe baseline, describe experiment settings

plot accuracy of 3 methods with different training size

### 4.2 Named Entity Classification

1) describe task and dataset

2) describe baseline DLCoTrain

3) experiment with NEC data

4) experiment DLCoTrain and tumbl on ppattach set

5) analysis

### 4.3 AAN Terminology Extraction

Not sure about this, if time is limited, show only preliminary results, no comparison w/ other methods.

## 5 Related Work

1) Relation to Zhu's method harmonic function and Gaussian fields

– distribution assumption

Zhu's method assumes Gaussian fields, the propability distribution of random walk is a continuous Gaussian distribution on the reverse of the distance between any two nodes. The graph is fully connected. It is a good model when the geometric distance is well defined.

In the bipartite graph model, the propability of one random step is propotional to the number of common features shared by the two example nodes. It is actually the dot product of two examples (recall that an example is a vector of dimention $m$). If the number of features connected to every example is same, i.e. every example has same magnitude, like in the pp attachment case, then it's also cosine). The random walk propability is a discrete distribution.

– advantage of tumbl

graph construction is cheap: $O(nm)$, Zhu's method needs to compute a $n \times n$ weight matrix, where each entry contains calculation of distance, so the complexity is $O(n^2 m)$

2) random walk/ label propagation applied to NLP tasks

Todo: check the related work cited in the word polarity paper e.g. Rao (2009) for sentiment classification

## 6 Conclusion

## References