



Learning Label Initialization for Time-Dependent Harmonic Extension

Amitoz AZAD

Acknowledgments



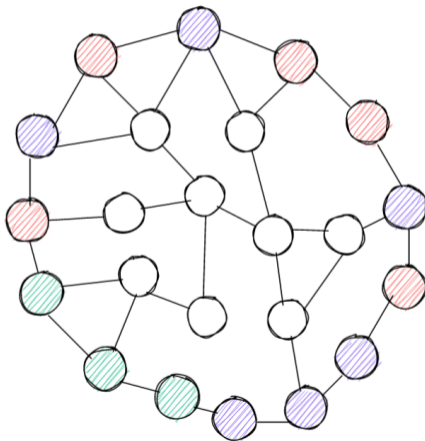
UNIVERSITÉ
CAEN
NORMANDIE



French National Research Agency

Problem

Harmonic Extension



$$\langle \text{blue hatched} \ \text{red hatched} \ \text{green hatched} \rangle \quad f(u) = g(u)$$

B.C.

$\langle \text{white} \rangle$
Interior

$$\triangle f(u) = 0$$

Inpainting

known pixels: $f(u) = g(u)$



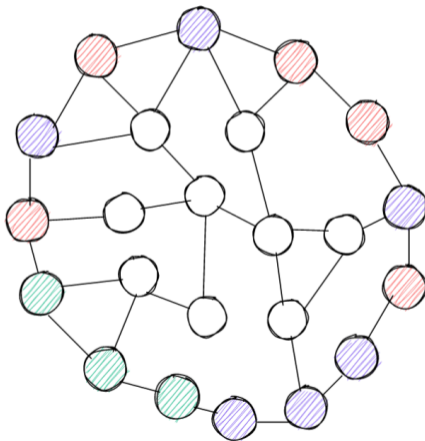
INPAINTED



unknown pixels: $\Delta f(u) = 0$

Problem

Harmonic Extension



$\langle \text{shaded nodes} \rangle \quad f(u) = g(u)$

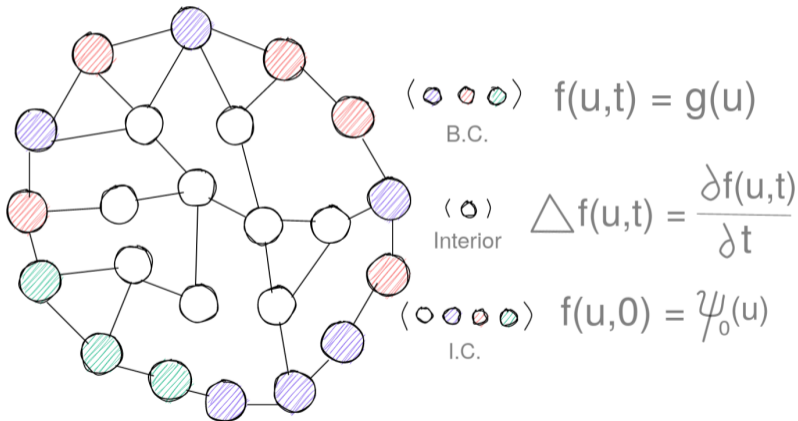
B.C.

$\langle \text{white nodes} \rangle$
Interior

$$\Delta f(u) = 0$$

Problem

Time-Dependent Harmonic Extension (TDHE)



Contributions

- Learning the initial condition of the PDE results in competitive performance.
- The proposed model can incorporate new labels without retraining.

Time-Dependent Harmonic Extension (TDHE)

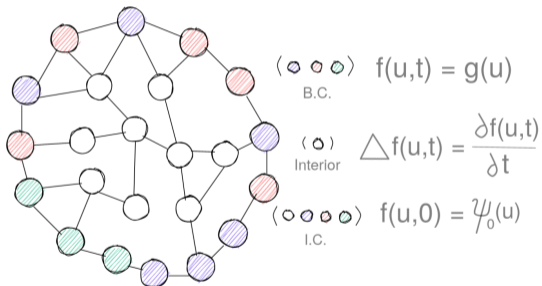


Table of Contents

1. Introduction
2. Background
3. Experiments

Variational Problem

$$\sum \|\nabla f(u)\|^2$$

Minimize **Dirichlet** energy of the whole graph,

- Convex problem.

$$f(u) = g(u)$$

given B.C.

PDE Problem

- Vector valued PDE.
- Symmetric graph.

$$\Delta f(u) = 0$$

In the interior

$$f(u) = g(u)$$

On boundary

Time-Dependent PDE

- Heat diffusion
non-homogenous
boundary condition.
- Similar to front
propagation
methods for
segmentation.

$$\Delta f(u, t) = \frac{\partial f(u, t)}{\partial t}$$

Diffusion in the interior

$$f(u, t) = g(u)$$

B.C.

$$f(u, 0) = \psi_0(u)$$

I.C.

Level-Sets Based Segmentation

$$\frac{\partial \phi(u, t)}{\partial t} = \kappa(u) \|\nabla_w \phi(u, t)\|$$

Front evolution

$$\phi(u, 0) = \phi_0(u)$$

Initial condition



Link with Zhu's[2002] Label Propagation

$$\nabla f(u, v) := \sqrt{w(u, v)}(f(v) - f(u))$$

Define gradient.

$$\operatorname{div}(F)(u) := \sum_{v \in N(u)} \sqrt{w(u, v)}(F(u, v) - F(v, u))$$

Define divergence.

$$\Delta f(u) := \frac{1}{2} \operatorname{div}(\nabla f)(u)$$

Use the Laplacian in the harmonic extension.

Table of Contents

1. Introduction
2. Background
3. Experiments

Research Question



How does the solution of TDHE compare with the GCN [Kipf] for semi-supervised node classification on citation networks ?

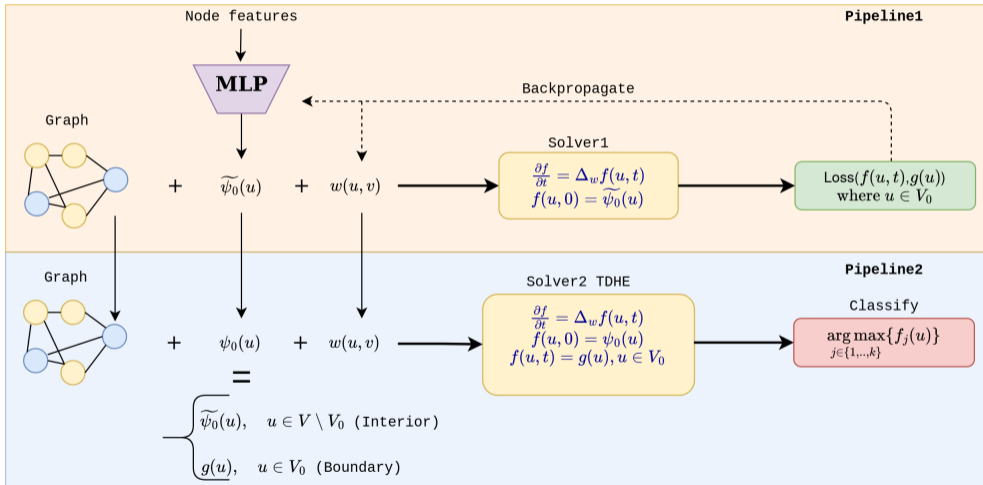
Citation Networks Results

	Cora	Citeseer	Pubmed
GCN [Kipf]	81.5	70.3	79.0
TDHE default ψ_0	72.5	49.7	72.5

Research Question

- ✓ How to make the solution of TDHE at par with the GCN by utilising the node features ?

Proposed Model



Citation Networks Results

	Cora	Citeseer	Pubmed
GCN [Kipf]	81.5	70.3	79.0
TDHE default ψ_0	72.5	49.7	72.5
TDHE learned ψ_0	82.9	72.2	81.1

Citation Networks Results

	Cora	Citeseer	Pubmed
GCN [Kipf]	81.5	70.3	79.0
TDHE default ψ_0	72.5	49.7	72.5
TDHE learned ψ_0	82.9	72.2	81.1
TDHE learned ψ_0 & w	83.5	72.4	81.9

Research Question



How to use the new labels without retraining the model ?

How to use the new labels ?

- Update the boundary condition: let the new labels be included in the boundary.
- Solve TDHE using the learned front ψ_0 and the updated boundary condition.

Citation Networks Results

	Cora	Citeseer	Pubmed
GCN [Kipf]	81.5	70.3	79.0
TDHE default ψ_0	72.5	49.7	72.5
TDHE learned ψ_0	82.7	72.2	81.1
TDHE learned ψ_0 & w	83.5	72.4	81.9
TDHE inc. val (ψ_0, w)	87.2	73.3	83.5

Citation Networks Results (time)

	Cora	Citeseer	Pubmed
GCN [Kipf]	81.5 4.0s	70.3 7.0s	79.0 38s
TDHE default ψ_0	72.5 0.2s	49.7 0.2s	72.5 0.2s
TDHE learned ψ_0	82.7 3.8s	72.2 7.4s	81.1 31s
TDHE learned ψ_0 & w	83.5 3.9s	72.4 7.7s	81.9 34s
TDHE inc. val (ψ_0, w)	87.2 0.2s	73.3 0.2s	83.5 0.2s

More Results

Model	Cora	Citeseer	Pubmed	Photo	CountyFB	FMAAsia
GCN	$81.5 \pm -\dagger$	$70.3 \pm -\dagger$	$79.0 \pm -\dagger$	93.4 ± 0.6	76.4 ± 1.4	85.9 ± 0.7
GAT	$83.0 \pm 0.7^\dagger$	$72.5 \pm 0.7^\dagger$	$79.0 \pm 0.3^\dagger$	94.1 ± 0.7	76.3 ± 1.6	87.0 ± 0.8
MoNet	$81.7 \pm 0.5^\dagger$	70.3 ± 0.9	$78.8 \pm 0.4^\dagger$	94.3 ± 0.3	79.4 ± 1.5	86.1 ± 1.0
C&S (MLP)	80.0 ± 0.9	67.3 ± 1.3	75.9 ± 1.9	94.2 ± 0.5	82.2 ± 1.2	87.3 ± 0.7
CGNN(w/o)	$82.7 \pm 0.7^\dagger$	$71.5 \pm 1.1^\dagger$	$81.7 \pm 0.5^\dagger$	93.3 ± 0.2	80.5 ± 1.4	85.3 ± 0.4
CGNN(w/)	$82.8 \pm 0.6^\dagger$	$72.1 \pm 0.8^\dagger$	$81.7 \pm 0.8^\dagger$	92.2 ± 0.7	81.9 ± 0.9	85.4 ± 0.3
TDHE	72.5 ± 0.0	49.7 ± 0.0	72.5 ± 0.0	91.3 ± 0.5	81.4 ± 1.8	86.7 ± 0.8
TDHE learned ψ_0	82.9 ± 0.7	72.2 ± 0.6	81.1 ± 0.9	94.3 ± 0.5	82.4 ± 1.5	87.0 ± 0.9
TDHE learned ψ_0 & w	83.5 ± 0.7	72.4 ± 0.7	81.9 ± 0.9	94.4 ± 0.5	82.6 ± 1.8	87.1 ± 0.8
TDHE inc. val	79.5 ± 0.0	60.4 ± 0.0	78.4 ± 0.0	92.4 ± 0.6	82.8 ± 1.4	87.2 ± 0.7
TDHE inc. val, (ψ_0, w)	87.2 ± 0.0	73.3 ± 0.0	83.5 ± 0.0	95.5 ± 0.5	85.3 ± 1.4	88.6 ± 0.8

Table: Test accuracy over different datasets. \dagger Values taken from the original papers. Coloring scheme: red-1st, blue-2nd, green-3rd.

Key take-homes

- ✓ Parameter t is important.
- ✓ Learning ψ_0 results in the competitive performance of the PDE.
- ✓ Learning the weights further yields minor gains.
- ✓ The proposed approach incorporates new labels without retraining.

Thank You :)