KU LEUVEN

Churn Prediction using Dynamic RFM-Augmented node2vec

Sandra Mitrović, Jochen de Weerdt, Bart Baesens & Wilfried Lemahieu

Department of Decision Sciences and Information Management, KU Leuven

18 September 2017, DyNo Workshop, ECML 2017
Skopje, Macedonia

Outline

- Introduction
- Motivation
- Methodology
- Experimental evaluation
- Results
- Conclusion
- Future work

Introduction

Churn prediction (CP)

- Predict which customers are going to leave company's services
  - Still considered as topmost challenge for Telcos (FCC report, 2009)
    - Due to acquisition/retention cost imbalance

- Different types of data used for CP
  - Subscription, socio-demographic, customer complaints etc.
  - More recently: Call Detail Records (CDRs)

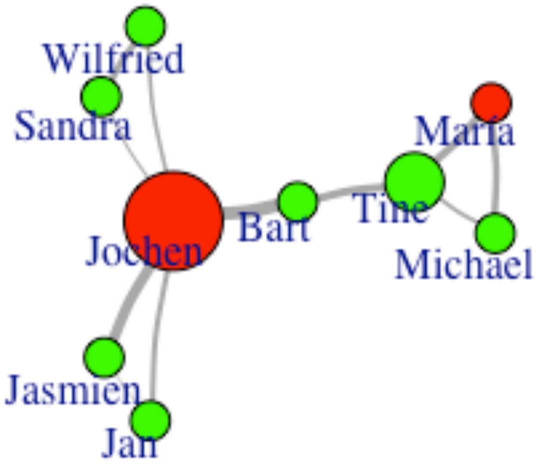
- CDRs -> call graphs

Date	Call Duration(sec)	Caller Number	Callee Number
2008-09-02 20:44:19	34	24002937	24997766
2008-09-02 20:42:56	26	24002937	24997766
2008-09-02 20:39:05	29	24002937	24997766
2008-09-02 20:38:06	24	24002937	24997766

Call graph featurization

Extracting informative features from (call) graphs

- An intricate process, due to:
  - Complex structure / different types of information
    - Topology-based (structural)
    - Interaction-based (as part of customer behavior)
      - Edge weights quantifying customer behavior
  - Dynamic aspect
    - Call graph are time-evolving
    - Both nodes and edges volatile
      - Churn = lack of activity



KU LEUVEN

Motivation

Problems identified (w.r.t. current literature)

- Not many studies account for **dynamic aspects of call networks**
  - Especially not **jointly with interaction and structural features**
    - Structural features are under-exploited
    - Due to high computational time in large graphs (e.g. betweenness centrality)
  - And **without using ad-hoc handcrafted features**
    - No featurization methodology
    - Dataset dependent

Our goal

- Performing holistic featurization of call graphs
  - Incorporating both interaction and structural information
  - Avoiding/reducing feature handcrafting
  - While also capturing the dynamic aspect of the network

Methodology

How do we address these goals?

G1: Incorporating both interaction and structural information →	Devise different operationalizations of RFM features and novel RFM-augmented call graph architectures
G2: Avoiding/reducing feature handcrafting →	Opt for representation learning
G3: Capturing the dynamic aspect of the network →	Slice original network into weekly snapshots

Integrating interaction and structural information

Interactions

(current literature)

- Usually delineated with RFM (Recency,Frequency,Monetary) variables
  - Benefits:
    - Simple
    - Yet still with good predictive power
  - Many different operationalizations
    - Different dimensions
    - Different granularities

Interactions

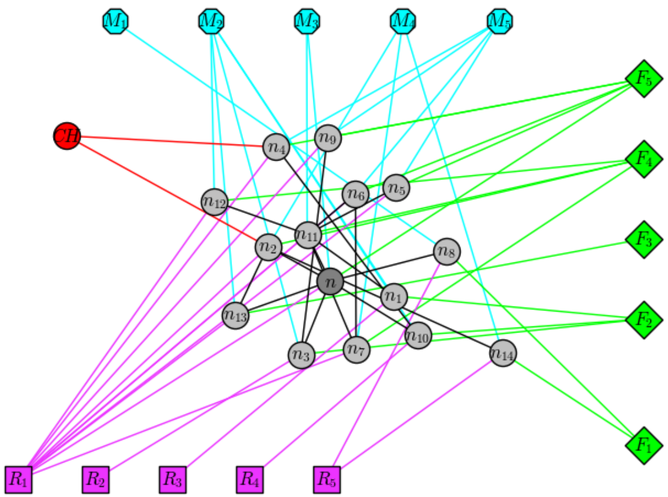
(this work)

- Summary RFM (RFM_s)
- Detailed RFM (RFM_d)
  - Direction & destination sliced:

    $X_{out\_h},\ X_{out\_o},\ X_{in},\ X \in \{R,F,M\}$
- Churn RFM (RFM_{ch})
  - Only w.r.t. churners

RFM-Augmented networks

- Original topology extended
  - By introducing artificial nodes based on RFM
  - Structural information partially preserved
- Each of R, F, M partitioned into 5 quantiles
  - One artificial node assigned to each quantile
  - Interaction info embedded through extended topology

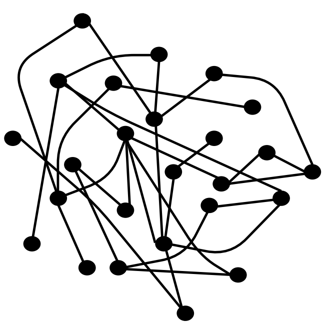


RFM features

- RFM_s
- $RFM_s \,||\, RFM_{ch}$
- RFM_d
- $RFM_d \,||\, RFM_{ch}$

+

Network topology



4 augmented networks

- AG_s
- AG_{s+ch}
- AG_d
- AG_{d+ch}

Representation learning

Node2vec

- Idea: Bring the representations of the words from the same context C close (borrowed from SkipGram)
  - Learn f, f: V -> R^d, d<< |V| s.t. max $\Sigma_{v \text{ in } V}$ log Pr(C_v | f(v))
- Definition of context in graph setting?
  - Neighborhoods/Random walks
    - Of which order? How to perform a walk?
- Flexible walks using additional parameters
  - Return parameter p
  - In-out parameter q
  - Coming from i, probability to transition

from j to k is: $\begin{cases} w_{jk}, \text{ if } d_{ik} = 1 \\ w_{jk}/p, \text{ if } d_{ik} = 0 \\ w_{jk}/q, \text{ if } d_{ik} = 2 \end{cases}$

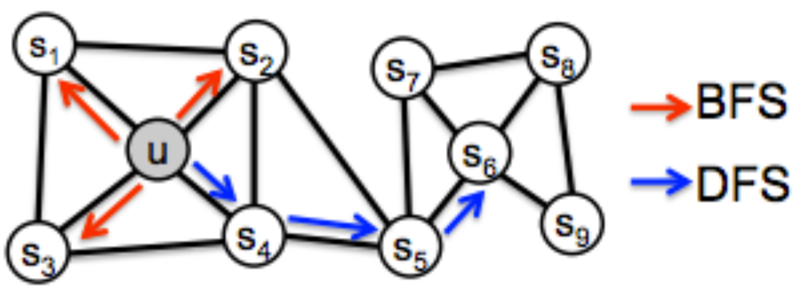


Figure source: Grover & Leskovec, 2016

Node2vec -> scalable node2vec

Node2vec

- Accounts both for previous and current node
- Additional parameters (p,q)
- To make walks efficient, requires precomputation of probability transitions:
    - On node level (1st time)
    - On edge level (successive)
    - Alias sampling used for efficient sampling
        - reduces O(n) to O(1)

However, does not scale well on large graphs!

(our case ~ 40M edges)

Scalable node2vec

- Accounts only for current node
- No additional parameters
- Requires precomputation of probability transitions only on node level
    - Alias sampling retained

Therefore, scales well even on large graphs!

Dynamic graphs

Different definitions (current literature)

- $G = (V, E, T)$
- $G = (V, E, T, \Delta T)$
- $G = (V, E, T, \sigma, \Delta T)$

Standard approach

- Consider several static snapshots of a dynamic graph

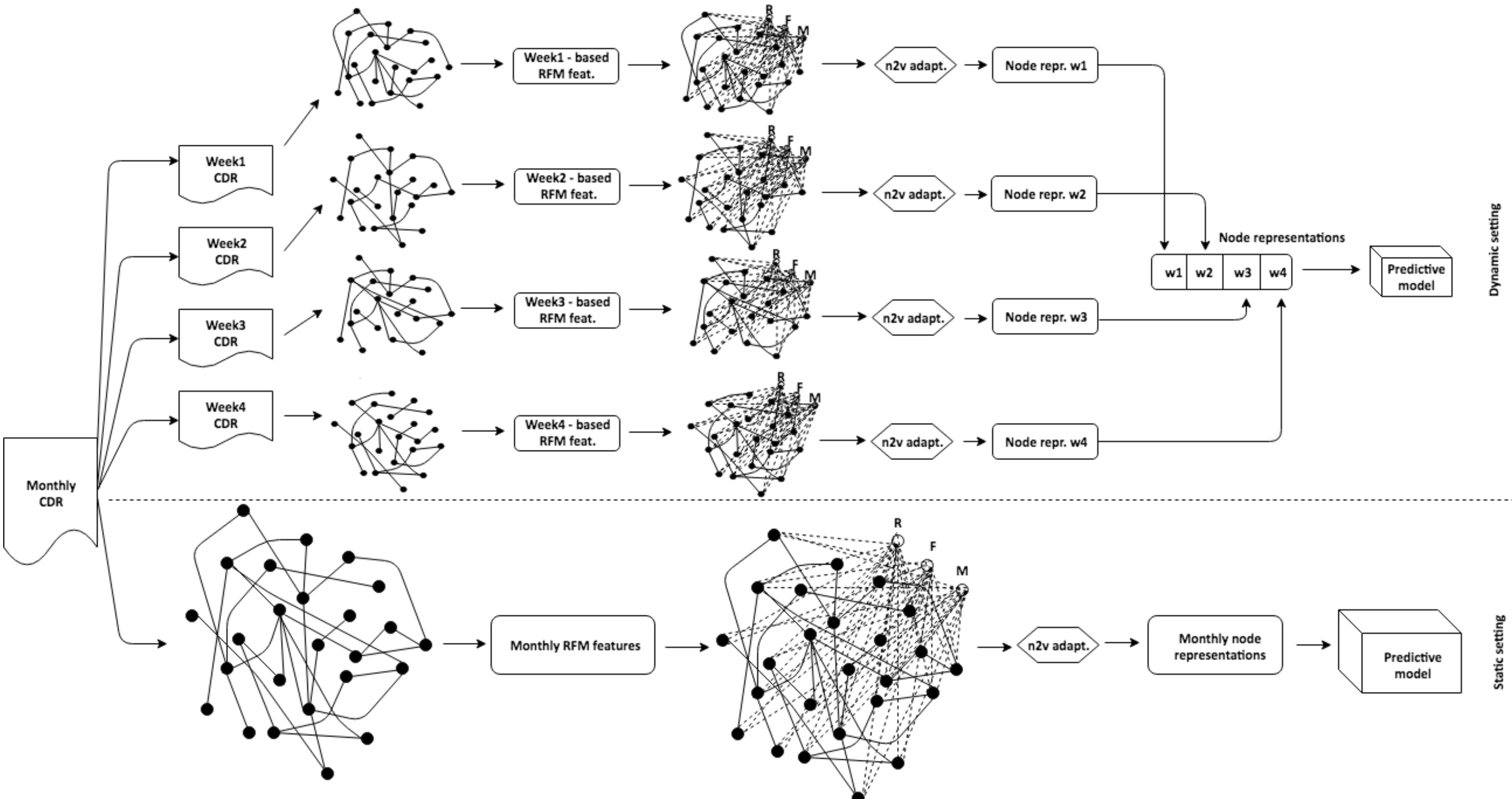
Our setting

- Monthly call graph $G = (V, E)$ ->

  Four temporal graphs $G_i = (V_i, E_i, w_i)$, $i = 1,..,4$

KU LEUVEN

Methodology – Graphical overview



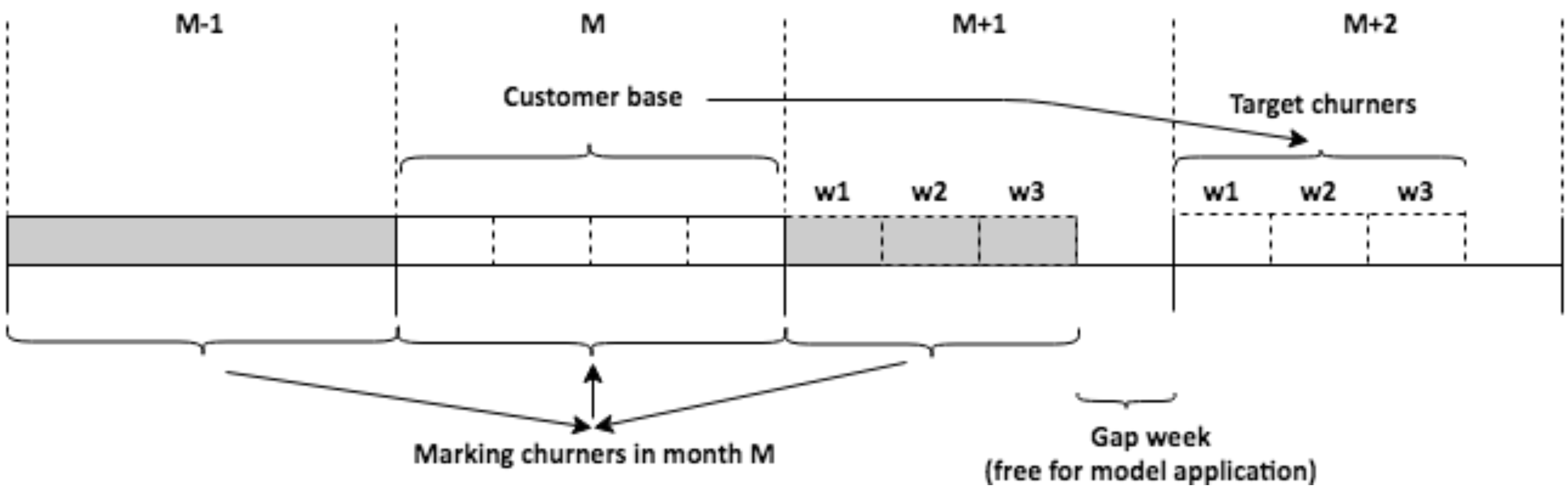
KU LEUVEN

Experimental Evaluation (1/2)

- One prepaid, one postpaid dataset
  - 4 months data (only CDRs)
- Undirected networks
- Model
  - Logistic regression with L_2 regul.

  (10-fold CV for tuning hyperparam.)
- Evaluation
  - AUC, lift (0.5%)

Parameter	Scalable node2vec
# walks	10
walk length	30
context size	10
# dimen.	128
# iterations	5



KU LEUVEN

Experimental Evaluation (2/2)

Research questions

- RQ1: Do features taking into account dynamic aspects perform better than static ones?
- RQ2: Do RFM-augmented network constructions improve predictive performance?
- RQ3: Does the granularity of interaction information (summary, summary +churn, detailed, detailed+churn) influence the predictive performance?

Experiments

- *RFM_{s} stat. vs. RFM_{s} dyn. vs. AG_{s} stat. vs. AG_{s} dyn. -> summary*
- *RFM_{s+ch} stat. vs. RFM_{s+ch} dyn. vs. AG_{s+ch} stat. vs. AG_{s+ch} dyn. -> summary+churn*
- *RFM_{d} stat. vs. RFM_{d} dyn. vs. AG_{d} stat. vs. AG_{d} dyn. -> detailed*
- *RFM_{d+ch} stat. vs. RFM_{d+ch} dyn. vs. AG_{d+ch} stat. vs. Ag_{d+ch} dyn. -> detailed*+churn

KU LEUVEN

Experimental results (1/2)

Prepaid

RFM	Static		Dynamic		Augmented network	Static		Dynamic	
	AUC	Lift	AUC	Lift		AUC	Lift	AUC	Lift
RFM_s	0.671	1.788	0.680	2.025	AG_s	0.680	**2.061**	**0.694**	2.013
RFM_{s+ch}	0.671	1.789	0.689	2.014	AG_{s+ch}	0.680	1.976	**0.705**	**2.331**
RFM_d	0.683	1.857	0.692	**2.063**	AG_d	0.678	1.898	**0.693**	2.019
RFM_{d+ch}	0.682	1.856	0.695	2.040	AG_{d+ch}	0.680	1.967	**0.702**	**2.316**

- RQ1 Answer: Dynamic better than static!
- RQ2 Answer: RFM-augmented networks improve predictive performance
- RQ3 Answer: Best performing interaction granularity is: summary+churn
  - Second best: detailed+churn

Experimental results (2/2)

Postpaid

RFM	Static		Dynamic		Augmented network	Static		Dynamic	
	AUC	Lift	AUC	Lift		AUC	Lift	AUC	Lift
RFM_s	0.741	3.367	0.743	3.403	AG_s	0.759	3.602	**0.768**	**3.919**
RFM_{s+ch}	0.741	3.369	0.758	3.858	AG_{s+ch}	0.760	3.553	**0.769**	**3.928**
RFM_d	0.750	3.750	0.757	3.874	AG_d	0.754	3.716	**0.764**	**3.908**
RFM_{d+ch}	0.750	3.751	**0.767**	3.885	AG_{d+ch}	0.755	3.720	0.764	**3.901**

- RQ1 Answer: Dynamic better than static!
- RQ2 Answer: RFM-augmented networks improve predictive performance
- RQ3 Answer: Best performing interaction granularity is summary+churn
  - Second best: summary

Conclusion

- We design **RFM-augmentations** of original graphs
  - Enable conjoining interaction and structural information
- We devise a **scalable** adaption of the original node2vec approach
  - Relaxing random walk generation and avoiding grid search tuning for two additional parameters
- Conducted experiments showcase the **performance benefits** which stem from taking into account the **dynamic** aspect
  - Also from exploiting RFM-augmented networks and learning node representations from these
- Novelty:
  - First work both in using (dynamic) node representations in CDR graphs for churn prediction and
  - First work in applying the RFM framework together with unsupervised and dynamic learning of node representations

Future research

- Attempt capturing call dynamics in a more sophisticated manner (e.g. the ordering of calls, their inter-event time distribution)
- Investigate the effect of different time granularities
- Explore whether prioritizing more recent dynamic networks improves performance

Thank you!

Questions?

Email: sandra.mitrovic@kuleuven.be

KU LEUVEN