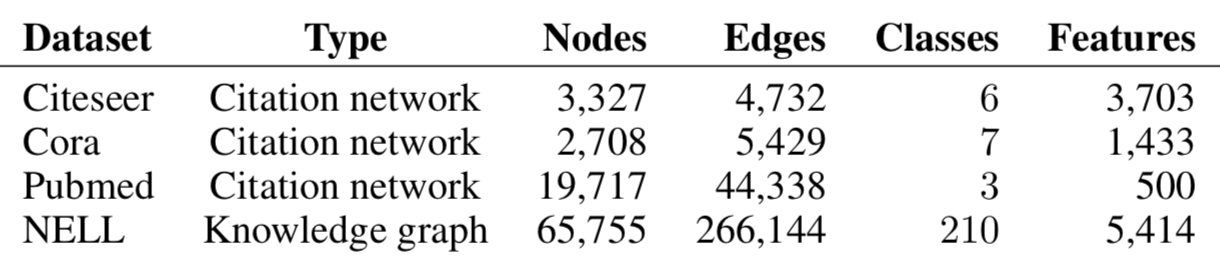
Thomas N. Kipf, Max Welling. 2017, [Semi-supervised Classification With Graph Convolutional Networks](https://arxiv.org/pdf/1609.02907.pdf), In *International Conference on Learning Representations (ICLR)*

**Code:**

<https://github.com/tkipf/gcn>

<https://github.com/tkipf/pygcn>

**Datasets:**

**1-** [Citeseer](https://github.com/kimiyoung/planetoid/tree/master/data)

**2-** [Cora](https://github.com/kimiyoung/planetoid/tree/master/data)

**3-** [Pubmed](https://github.com/kimiyoung/planetoid/tree/master/data)

**4-** [NELL](http://www.cs.cmu.edu/~zhiliny/data/nell_data.tar.gz)

**Datasets: Details**

In the citation network datasets—Citeseer, Cora and Pubmed—nodes are documents and edges are citation links. Label rate denotes the number of labeled nodes that are used for training divided by the total number of nodes in each dataset. NELL is a bipartite graph dataset extracted from a knowledge graph with 55,864 relation nodes and 9,891 entity nodes.

**1- Citation Networks**

* The datasets contain sparse bag-of-words feature vectors for each document and a list of citation links between documents.
* Citation links are treated as (undirected) edges and construct a binary, symmetric adjacency matrix A.
* Each document has a class label. For training, only 20 labels per class are used, but all feature vectors.

**2- NELL**

* NELL is a dataset extracted from the knowledge graph. A knowledge graph is a set of entities connected with directed, labeled edges (relations).
* Separate relation nodes r1 and r2 are assigned for each entity pair (e1,r,e2) as (e1,r1) and (e2,r2).
* Entity nodes are described by sparse feature vectors.
* The number of features in NELL is extended by assigning a unique one-hot representation for every relation node, effectively resulting in a 61,278-dim sparse feature vector per node.
* Only a single labeled example per class in the training set.
* A binary, symmetric adjacency matrix from this graph is constructed by setting entries Aij = 1, if one or more edges are present between nodes i and j.

**3- Random Graphs**

* For a dataset with N nodes, a random graph assigning 2N edges uniformly at random is created.
* The identity matrix IN is taken as input feature matrix X.
  + Implicitly taking a featureless approach where the model is only informed about the identity of each node, specified by a unique one-hot vector.
* Dummy labels Yi = 1 for every node are added.

**Baselines**

**1- Semi-Supervised Node Classification**

1. Label propagation (LP) (Zhu et al., 2003)
2. Semi-supervised embedding (SemiEmb) (Weston et al., 2012)
3. Manifold regularization (ManiReg) (Belkin et al., 2006)
4. Skip-gram based graph embeddings (DeepWalk) (Perozzi et al., 2014)
5. Planetoid (Yang et al., 2016)
6. Iterative classification algorithm (ICA) (Lu & Getoor, 2003)

**2- Propagation Model**

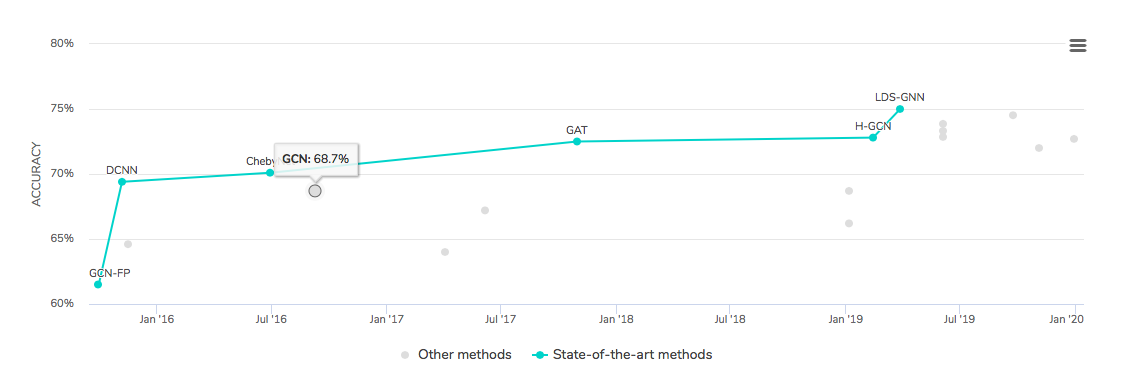
1. Chebyshev filter
2. 1st-order model
3. Single parameter
4. Renormalization trick
5. 1st-order term only
6. Multi-layer perceptron

**Upper bounds**

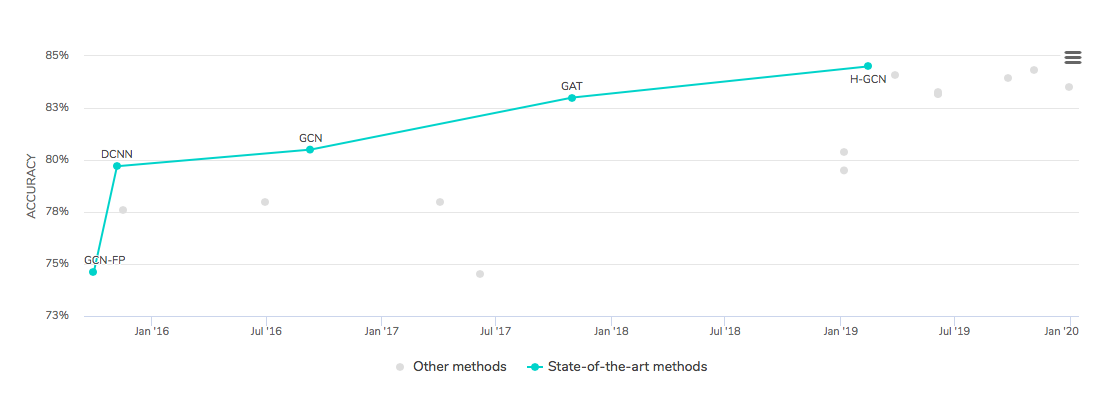
Not given.

**State-of-the-art**

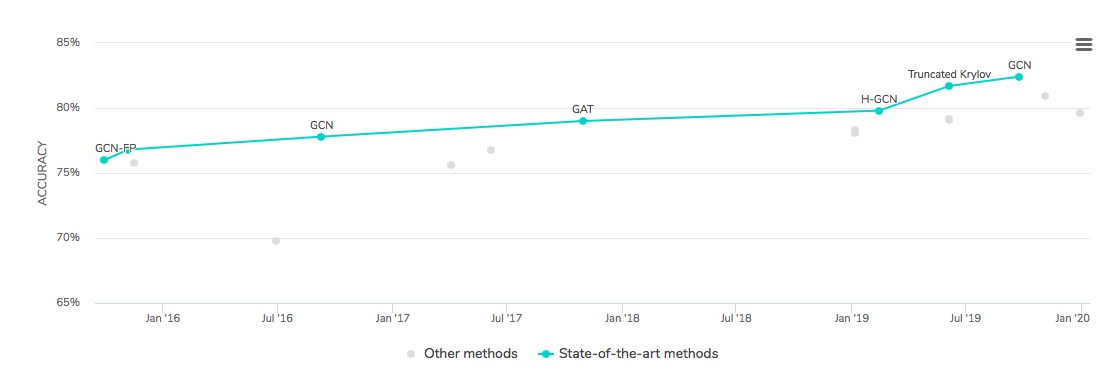
**1- Citeseer**

****

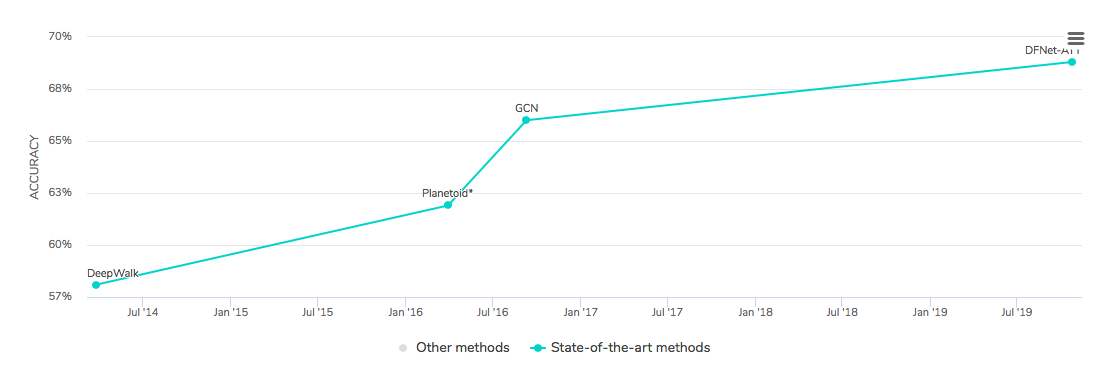
**2- Cora**

****

**3- PubMed**

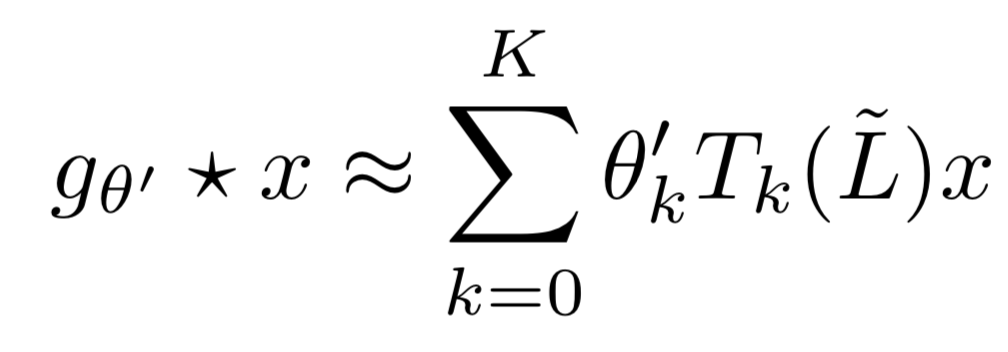
****

**4- NELL**

****

**Ablation Study**

The authors start with a baseline system and add these ideas one at a time to show what the impact is.

The definition of a convolution of a signal x with a filter gθ′:

* Limited the layer-wise convolution operation to K = 1

The effects:

* To make it a function that is linear w.r.t. L and therefore a linear function on the graph Laplacian spectrum.
* It is not limited to the explicit parameterization given by, e.g., the Chebyshev polynomials.
* It can alleviate the problem of overfitting on local neighborhood structures for graphs
* It allows us to build deeper models, a practice that is known to improve modeling capacity on a number of domains
* Approximated λmax ≈ 2

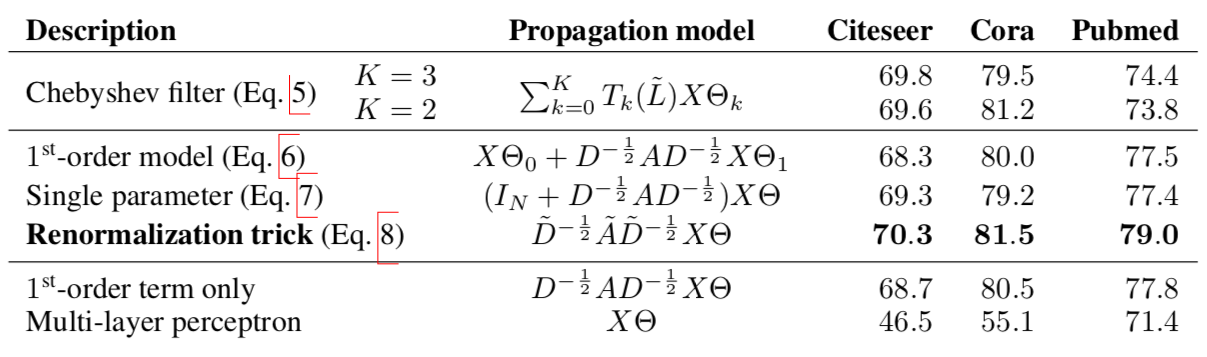
The effect:

* Successive application of filters of this form then effectively convolves the kth-order neighborhood of a node, where k is the number of successive filtering operations or convolutional layers in the neural network model.
* Constrained the number of parameters further

The effect:

* Addressed overfitting and to minimize the number of operations per layer.
* Renormalization trick

The effect:

* Avoided the numerical instabilities and exploding/vanishing gradients when used in a deep neural network model due to repeated application of the convolution operator.

**Supplementary**

[CS224W Analysis of Networks Mining and Learning With Graphs](http://snap.stanford.edu/class/cs224w-2018/)