To Appear as Full Paper at

CIKM 2017 SINGAPORE

A Two-Stage Framework for Computing Entity Relatedness in Wikipedia

Marco Ponza, Paolo Ferragina and Soumen Chakrabarti

University of Pisa *IIT Bombay*

Google NLP Summit 2017

Motivations

Proliferation of the usage of *Knowledge Graphs:*

- Retrieval of Information (Bordino, WSDM '13), (Cornolti, WWW '16)
- Entity Linking (Meij, WSDM '12), (Piccinno, SIGIR '14), (Ganea, WWW '16)
- Document Clustering, Classification and Similarity (Scaiella, WSDM '12), (Vitale, ECIR '12), (Ni, WSDM '16)

Need for computing *relatedness* between *entities!*

Contributions

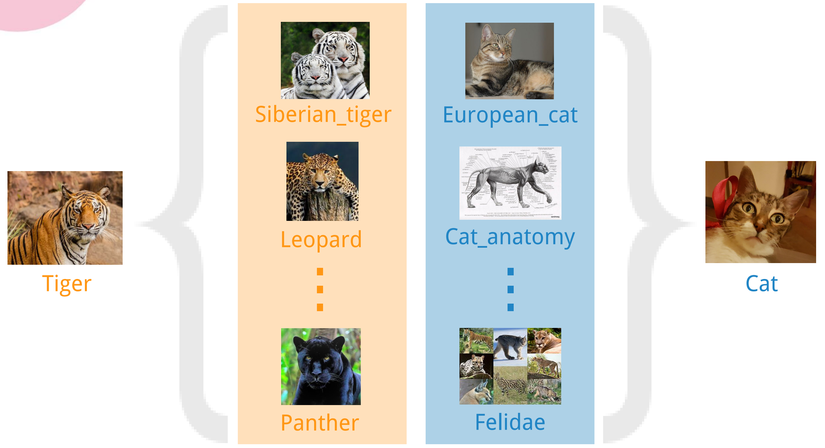
1. A new entity-relatedness dataset WiRe, comprising judgments by human experts on 503 pairs of Wikipedia entities
2. Intrinsic evaluation of all recent relatedness measures:
   - Personalized PageRank (Haveliwala, WWW '02)
   - Link Prediction (Liben-Nowell, JAIST '07)
   - Word and Document Similarity (Gabrilovich, IJCAI '07)
   - Word2Vec (Mikolov, NIPS '13)
   - CoSimRank (Rothe, ACL '14)
   - *...and more in the paper!*

   over the new Wire and the known WikiSim (Milne, AAAI '08) datasets
3. A new efficient Two-Stage Framework for relatedness computation:
   - Configurable joint framework without any need of feature engineering
   - Improvements is more than 5% with peaks of 7% on WiRe
4. Extrinsic evaluation of the new framework on entity linking
5. Publicly available datasets and algorithms

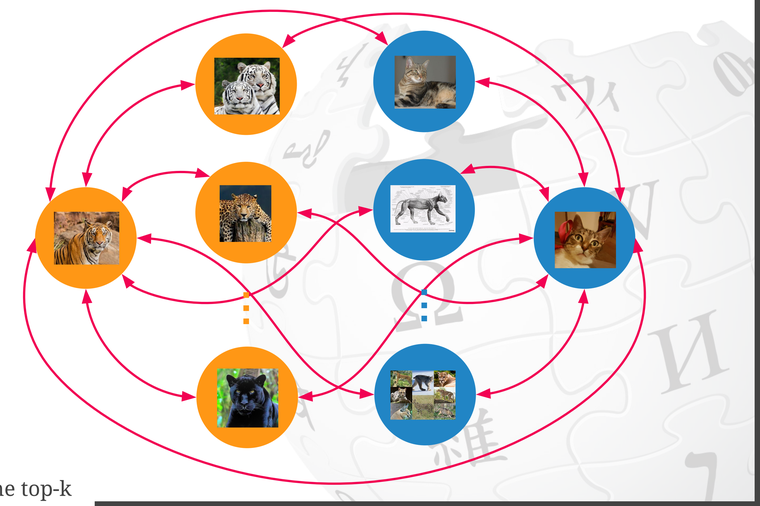
Our Two-Stage Framework

1 First Stage: Creation of a Wikipedia Subgraph

Choosing nodes of the subgraph



Creation of edges & their weights



Description of the First Stage. A Wikipedia subgraph is created by retrieving the top-k most related entities to the query entities *(i.e. choosing nodes of the subgraph)* and by subsequently linking them in a small and weighted sparse graph *(i.e. creation of edges & their weights)*. Both the top-k retrieval and weighting schemes are fully configurable with a various set of algorithms, such as ESA (Gabrilovich, IJCAI '07), Milne&Witten (Milne, AAAI '08), DeepWalk (Perozzi, KDD '14), Entity2Vec (Ni, WSDM '16).

2 Second Stage: Computing Relatedness

The relatedness between the two query entities is computed by running CoSimRank (Rothe, ACL '14) over this *small* and *weighted* graph. The overall computation (i.e. first + second stages) is *fast and can be performed at query time*.

Efficiency

By carefully optimizing a few steps (details in the paper) the framework turns out to be space-efficient and computationally lightweight!

	Uncompressed	Compressed
Space	5 GB	445 MB
Average Time	0.5 ms	3 ms

Experiments

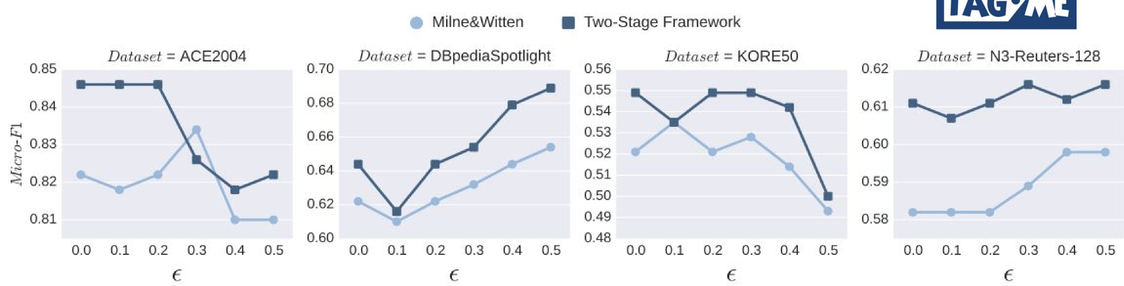
Intrinsic Evaluation over WikSim & WiRe Datasets

Method	WikiSim			WiRe			AVG
	Pearson	Spearman	Harmonic	Pearson	Spearman	Harmonic	
ESA	0.61	0.72	0.67	0.60	0.63	0.62	0.645
Milne&Witten	0.62	0.65	0.63	0.77	0.69	0.72	0.675
DeepWalk	0.71	0.70	0.71	0.74	0.68	0.71	0.710
Entity2Vec	0.68	0.70	0.69	0.74	0.70	0.72	0.705
Two-Stage Framework	**0.74**	**0.75**	**0.74**	**0.83**	**0.75**	**0.79**	**0.765**

Fair and comprehensive comparison of all relatedness methods present in the recent literature and properly adapted to our context (more experiments in the paper).

Extrinsic Evaluation: Entity Linking

Improvement of TAGME



We replaced the relatedness method used by TagMe (i.e. Milne&Witten) with our Two-Stage Framework. Our relatedness measure not only improves TagMe, but also makes it more insensitive to choices of the ε-parameter in TagMe.