Predicting Adverse Drug-Drug Interactions with Neural Embedding of Semantic Predications

Hannah A Burkhardt[1], Devika Subramanian[2], Justin Mower[2], Trevor Cohen[1]

[1] University of Washington, Seattle, WA, [2] Rice University, Houston, TX

UW Medicine
UW SCHOOL OF MEDICINE
BIOMEDICAL INFORMATICS AND MEDICAL EDUCATION

Background

- **Drug-drug interactions (DDIs)** are an important patient safety issue. Polypharmacy is more common than ever: 66.8% of >65 y.o.'s took ≥3 prescription medications in 2014 (National Center for Health Statistics 2017).
- Exhaustive pre-market investigation of possible DDIs is not feasible
- Fast and accurate **computational pharmacovigilance systems** are needed
- The DDI prediction problem can be framed as a **link prediction problem, where drugs are nodes and side effects are edges of different types**, e.g. gastric ulcers or excessive bleeding (Figure 1); then predict which links are likely to exist
- **Decagon** (Zitnik, Agrawal, and Leskovec 2018) is a **graph convolutional network** which takes this approach. Disadvantages of Decagon: High computational demand; embeddings that are difficult to re-use; and a huge volume of trainable parameters.

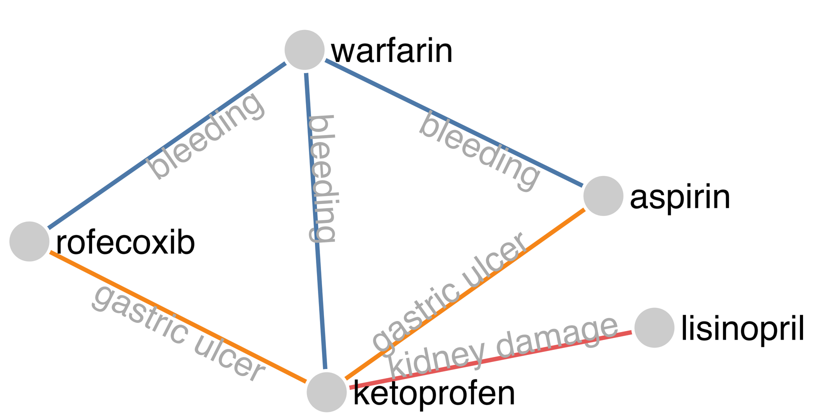

Figure 1: Graph representation of drug-drug interactions

Embedding of Semantic Predications (ESP)

- Alternatively, **ESP** (Cohen and Widdows 2017) can be used for this task.
- ESP generates **vector embeddings** for drugs and side effects from concept-relationship-concept triples called **predications**, using a neural network
- Makes use of **vector symbolic architectures**: composition (binding) can be used to create composite concepts (e.g. `aspirin-warfarin`) from component vectors.
- **Similar representations will be learned for similar concepts**: we can query the resulting vector space for concepts related to a given concept (or composition).
- For example, aspirin and warfarin are both blood thinners, so taking them together might cause excessive bleeding - the embedding of the compositional concept `aspirin-warfarin` is similar to the embedding for `bleeding`.

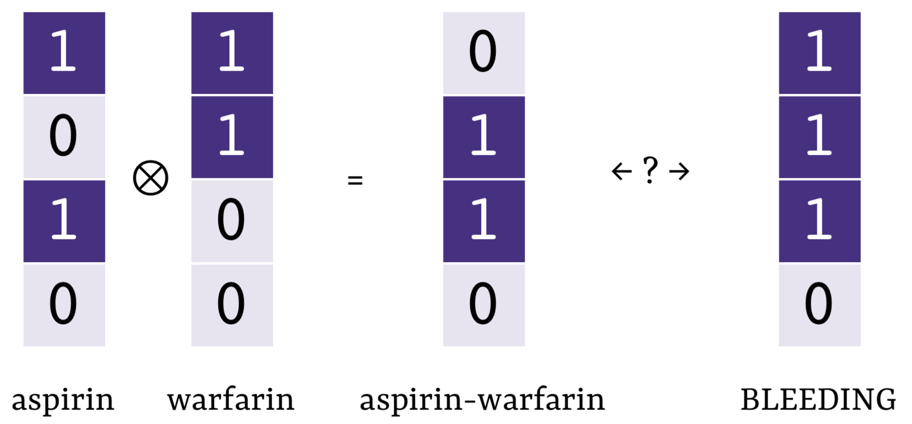

Figure 2: Compositional conceps result from binding vectors.

Methods

- We used the **same dataset** as Zitnik et al. and the **open-source Semantic Vectors package** (Widdows and Ferraro 2008) with 16k bit **binary vectors**
- Testing: the embeddings for the two drugs are bound, and the **similarity of the resulting vector with the target vector is scored**
- The vector space can be explored to qualitatively **evaluate whether the learned representations are meaningful**. We queried the vector space for drugs, side effects, and compositional concepts (Table 3)
- A UMAP projection shows the vector space with side effects colored by class.

Results

Table 1: Performance metrics with 95% confidence intervals.

	Mean AUROC	Mean AUPRC	Mean AP@50
ESP (4 epochs)	0.865 (0.804–0.927)	0.842 (0.765–0.919)	0.847 (0.692–1.000)
ESP (8 epochs)	**0.880** (0.827–0.933)	0.855 (0.786–0.924)	0.852 (0.701–1.000)
Decagon (4 epochs)	0.826 (0.681–0.971)	0.768 (0.636–0.900)	0.644 (0.378–0.909)
Decagon (published)	**0.872**	0.832	0.803

Table 2: Training time incl. setup time, and number of trainable bits of parameters.

	Total running time	Time per epoch	Trainable parameter bits
ESP	3.5 hours (2.4%)	0.8 hours (2.2%)	36 million (1.3%)
Decagon	144 hours	36 hours	2,800 million

Table 3: Example searches of the predicate vector space. The search returns the vectors that most similar to the query term, along with a similarity score

Cue	Result
`P(KIDNEY_FAILURE)` What side effects occur in similar drug combinations as kidney failure?	1.000: `KIDNEY_FAILURE` 0.949: `ACUTE_KIDNEY_FAILURE` 0.927: `RESPIRATORY_FAILURE` 0.923: `CARDIAC_FAILURE`
`S(aspirin)⊗C(warfarin)` What side effects might be caused by taking aspirin and warfarin together?	0.597: `FEMUR_FRACTURE` 0.582: `CARDIOMYOPATHY` 0.581: `THROMBOPHLEBITIS` 0.579: `BLOOD_DISORDER`

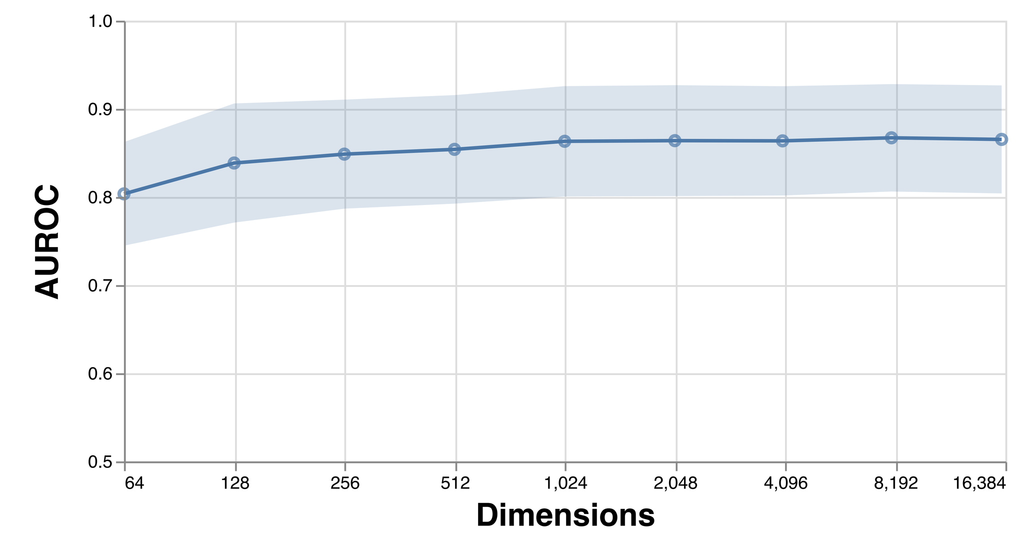

Figure 3: Mean AUROC (over 963 side effects) and 95% CIs by vector dimensionality

The mean area under the receiver operating characteristic curve over 963 side effects for different vector dimensionalities, created by truncating the 16,000 dimensional vectors, are shown in Figure 3.

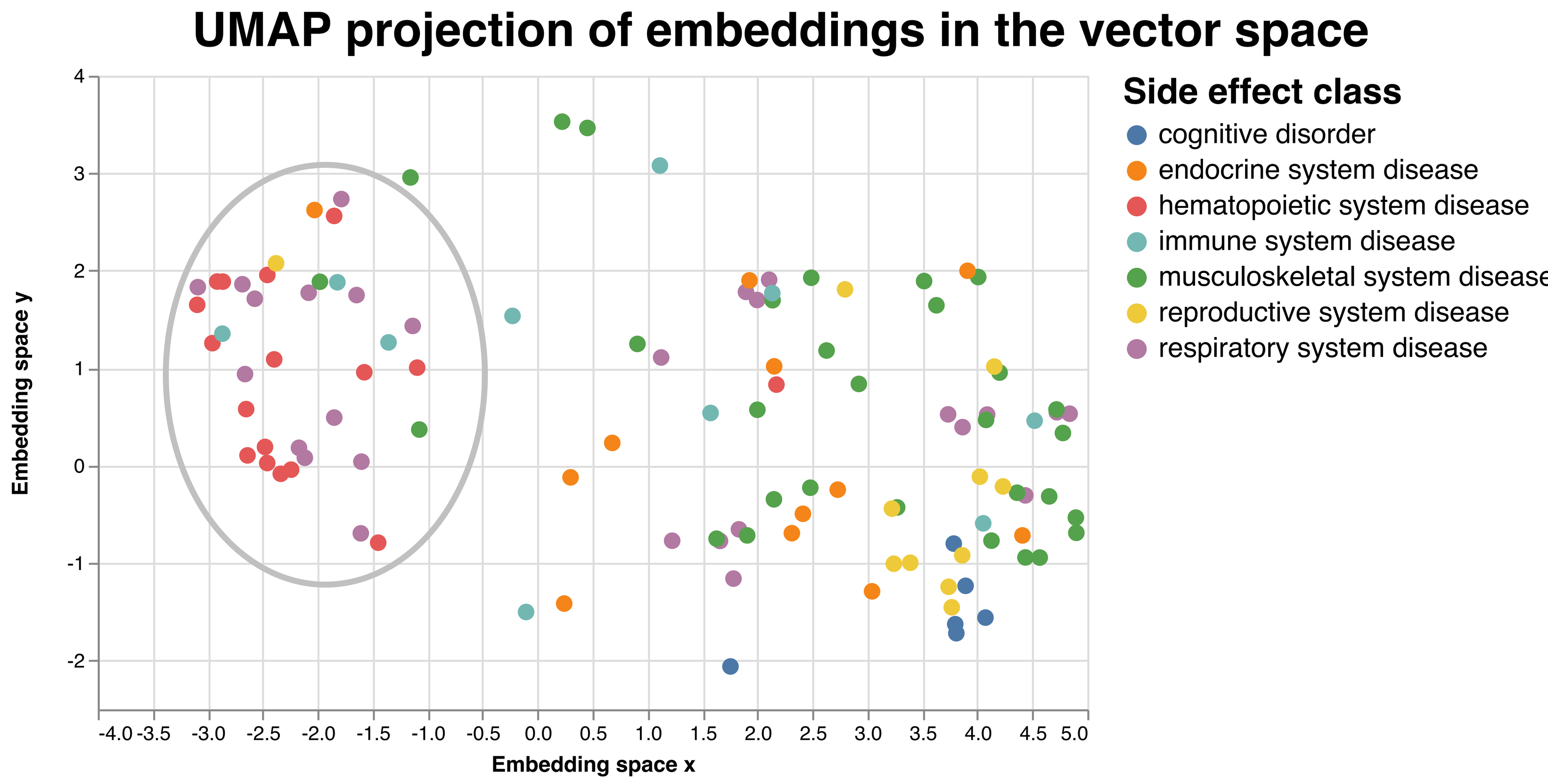

Figure 4: UMAP projection of several side effect groups in the vector space

Discussion & Conclusion

- **Embedding of Semantic Predications (ESP) can predict DDIs** slightly better than state-of-the-art ML systems **with 77 times fewer parameters**, though not statistically significantly so, and trains **43 times faster**
- High performance is seen **even at much smaller vector dimensionality**
- ESP produces reusable and **meaningful embeddings**
- Using curated DDI databases presents an opportunity for future work

Acknowledgements & References

This work is supported by a training fellowship (Grant No. 3T15 LM007442, Biomedical and Health Informatics Training Program) and U.S. National Library of Medicine Grant (2R01LM011563), Robust Inference from Observational Data with Distributed Representations of Conceptual Relations.

This work was submitted and accepted for the AMIA Annual Symposium 2019 (Burkhardt et al. 2019).

Burkhardt, Hannah A, Devika Subramanian, Justin Mower, and Trevor Cohen. 2019. "Predicting Adverse Drug-Drug Interactions with Neural Embedding of Semantic Predications." *To Appear in Proc AMIA Annu Symp 2019*.

Cohen, Trevor, and Dominic Widdows. 2017. "Embedding of semantic predications." *Journal of Biomedical Informatics* 68: 150–66. https://doi.org/10.1016/j.jbi.2017.03.003.

National Center for Health Statistics. 2017. "Health, United States, 2017." Centers for Disease Control; Prevention (CDC). https://www.cdc.gov/nchs/hus/contents2017.htm{\#}079.

Widdows, Dominic, and Kathleen Ferraro. 2008. "Semantic Vectors: A scalable open source package and online technology management application."

Zitnik, Marinka, Monica Agrawal, and Jure Leskovec. 2018. "Modeling polypharmacy side effects with graph convolutional networks." *Bioinformatics* 34 (13): i457–i466. https://doi.org/10.1093/bioinformatics/bty294.