# Pun generation

## [1] pun generation with surprise. Nanyun Peng

### Method: unsupervised sampling and generation

1. Pick up a sentence containing alternative word w\_a. Replace alternative word w\_a with pun word w\_p. The pronunciations of w\_a and w\_p are similar, which achieves local surprise.
2. Train a relatedness network 文本

   AI 生成的内容可能不正确。 in unsupervised manner 图示, 示意图

   AI 生成的内容可能不正确。. Sample candidate topic words from 文本

   AI 生成的内容可能不正确。 to foreshadow w\_p in the beginning of the sentence, which achieves global surprise. Note that the topic word will replace a word in the beginning of the sentence.
3. Guarantee that the finally selected topic word is type-consistent with the deleted word. Type-consistency is guaranteed based on WordNet path similarity.
4. Adopt a smoother to improve grammaticality.

* He He, Nanyun Peng, and Percy Liang. 2019. [Pun Generation with Surprise](https://aclanthology.org/N19-1172/). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1734–1744, Minneapolis, Minnesota. Association for Computational Linguistics.

## [2] a neural approach to pun generation.

### Method: unsupervised sampling and generation

1. Formulate and develop probability distribution with joint model and highlight model. The joint model outputs a joint score distribution 图片包含 文本

   AI 生成的内容可能不正确。. The highlight model outputs an associative word distribution 文本

   AI 生成的内容可能不正确。, where 文本

   AI 生成的内容可能不正确。 and 文本

   AI 生成的内容可能不正确。. The role of joint model is to guarantee a pun word with two different semantics meaning can be fitted into the generated sentence. The role of highlight model is to guarantee the two different semantics meaning of the pun word can be reminded and revealed by associative words to achieve surprise.
2. The above two models together form the final probability distribution 文本

   AI 生成的内容可能不正确。.
3. Adopt multinomial sampling to sample words from the final probability distribution. The sequentially sampled words form the pun generation.

* Zhiwei Yu, Jiwei Tan, and Xiaojun Wan. 2018. [A Neural Approach to Pun Generation](https://aclanthology.org/P18-1153/). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1650–1660, Melbourne, Australia. Association for Computational Linguistics.