Character Jacobian: modeling Chinese character meanings with deep learning models

Yu-Hsiang Tseng, Shu-Kai Hsieh
Graduate Institute of Linguistics
National Taiwan University

@COLING 2022
Gyeongju, Republic of Korea

In this study

- We propose a non-linear transformation model for Chinese compounding.
- It predicts real words' embeddings from their constituents, helps analyze the behavioral data of pseudowords, and models the characters' polysemous behaviors with the the Jacobian matrices.
- The results suggest we could study the compounds with deep learning models.

OUTLINE

1. (Chinese) Morphology and compounding
2. Nonlinear Transformation Model (Notch) for compounding
3. Analyzing pseudowords' behavioral data
4. Examining the characters' meanings with Character Jacobians
5. Conclusion

Compounding

- Compounding is a productive and prevalent word formation process in many languages. (Jackendoff, 2002)
- Compounds are loosely defined as forming words with two (or more) constituents.
- To determine the meanings of the compounds and the relations between the constituents is challenging:
    - airplane / airport

Compounding in Chinese

- Chinese words are composed of one or more characters, many of which have their own meanings.
- That is, most Chinese words may be considered compounds.
- Nearly all Chinese characters are polysemous.
    - 長老 zhǎng lǎo "*senior-elder, elder*"
    - 老師 lǎo shī "*prefix-teacher, teacher*"
    - 師法 shī fǎ "*learn-model, model after*"

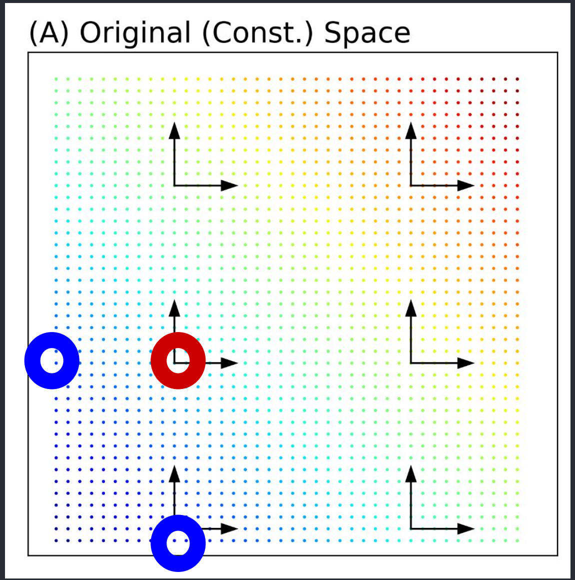
Modeling compounds

- One challenge of studying compounds is modeling the relations between the compound semantics and constituent semantics.
- We operationalize semantics with word vectors (Mikolov et al., 2013).
- Is the meaning of "airport" the composite of the meaning of "air" and the meaning of "port"?
- $v(\text{airport}) = v(\text{air}) + v(\text{port})$

Additive model

- v(airport) = v(air) + v(port)
  - v(air) := [5, 0]
  - v(port) := [0, 8]
  - v(airport) = [5, 8]
- The vectors are transformed by two matrices and added together

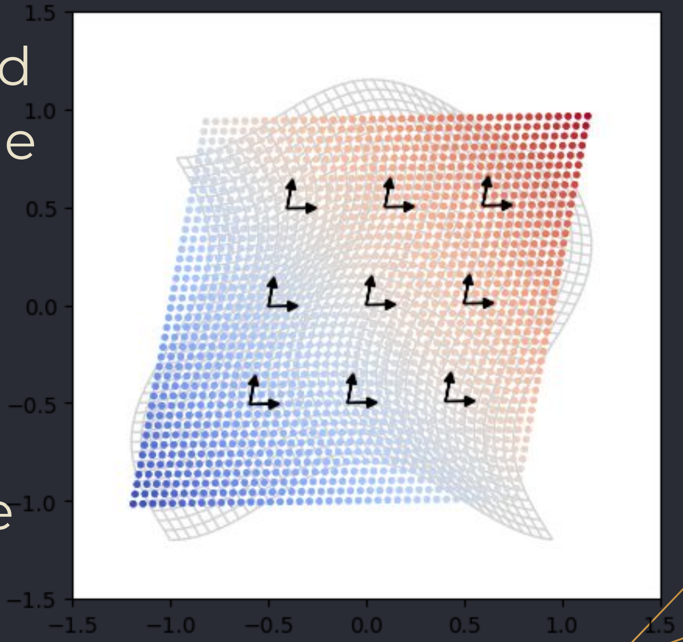
$$v_{\text{air}} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + v_{\text{port}} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$



Linear model

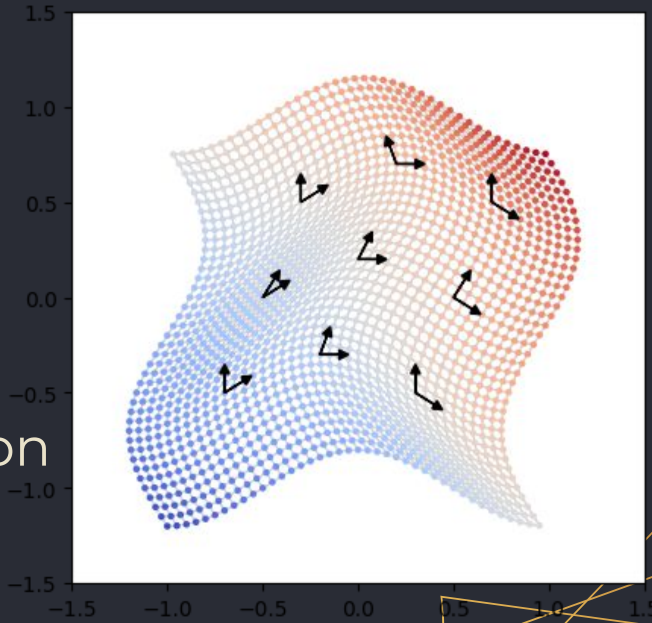
- The meaning of the constituents could be different when they are stand-alone words. (Libben, 2014; Gunther et al. 2021)
- We estimate M1 and M2 to transform the const. vectors. The transf. are the same everywhere in space.
- i.e., airport / airtight must be the same 'air' $v_{\text{air}} M_1 + v_{\text{port}} M_2$

$$v_{\text{air}} \begin{bmatrix} 1 & 0.01 \\ 0 & 0 \end{bmatrix} + v_{\text{port}} \begin{bmatrix} 0 & 0 \\ 0.19 & 1 \end{bmatrix}$$



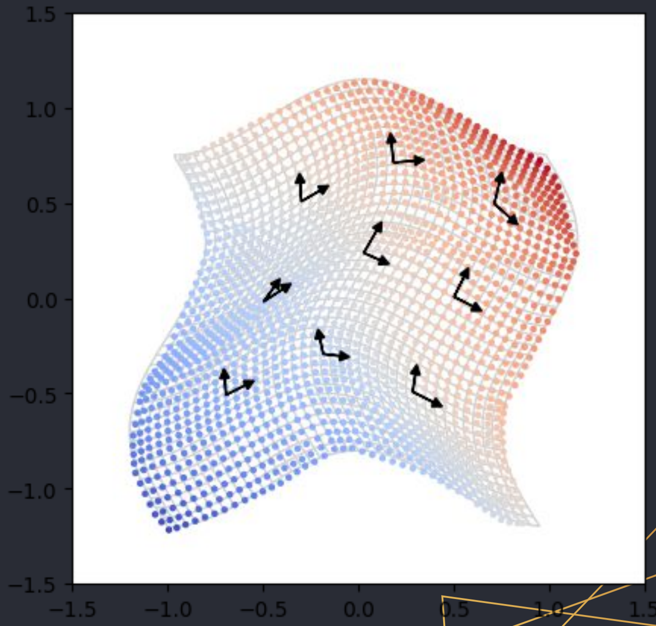
What if the relations are more complex ?

- What if the same constituent acts differently in different words? Just as the case in Chinese:
  - 握手 wò shǒu "*hold-hands*"
  - 鼓手 gǔ shǒu "*drum-er(suffix)*"
- The same constituent might need different transformations, depending on the word context.
- The transformation would be warped and highly non-linear.



What we are trying to do is...

- Build a non-linear model to capture these relations.
- We use BERT (Devlin et al., 2019) because it has pretrained weights based on large corpus, and might be a good start in modeling compounds.
- The non-linear transformation (Notch) model takes the constituents as inputs, and predicting the compounds' word vectors.



2. The Notch model

The model architecture

- The Notch model is based on a pre-trained BERT (bert-base-chinese), and an extra projection layer mapping the [CLS] token vectors (768d) to the word vectors (100d).

$$v_{\text{word}} = \text{Notch}(c_1 \ldots c_k)$$

- The model inputs are variable-length character sequences, and the outputs are word vectors.
- We used 490K Chinese word vectors to train the model. The word vectors (from Tencent AI lab) has 100 dimensions.

Evaluation on real words

- We compute the top-k accuracy on 10K held-out words: whether the predicted vectors are within the k-nearest neighbors of the true word vectors.

Len.	N	Top 1	Top 5	Top 10
1	162	.73	.85	.86
2	2,522	.63	.78	.81
3	2,123	.66	.79	.84
4	3,375	.75	.87	.90
$\geq$5	1,818	.57	.72	.77
All	10,000	.67	.80	.84

- The highest accuracies of the 4-char words might be partly due to the coarse-grained words included in Tencent embeddings. (乘坐高鐵, 乘坐 riding- 高鐵 high speed rail), which are more semantically transparent.

Some observations of the errors

- Some predictions, while not close to the true vectors, are semantically related:
  - 脱除 tuō chú: 去除, 消退, 卸去
    get rid of: discard; fade away; remove
- Predictions of opaque words might be (mis-)guided by the constituents' meanings:
  - 社交距离 shè jiāo jù lí: 彼此了解, 交流能力, 像朋友一样
    social distancing: mutual understanding; communication skills; (be) like friends

A short discussion of Notch model

- To some extent, the model learns to predict word meanings from its constituents.
- Does it imply more theoretical issue on compounding?
- The caveat is that we are operationalize semantics by word vectors. It is hard to tell the roles vector semantics are playing here.
- The bottom line is the model captures something about words and their constituents.

3. Analyzing behavioral data on pseudowords

Pseudowords

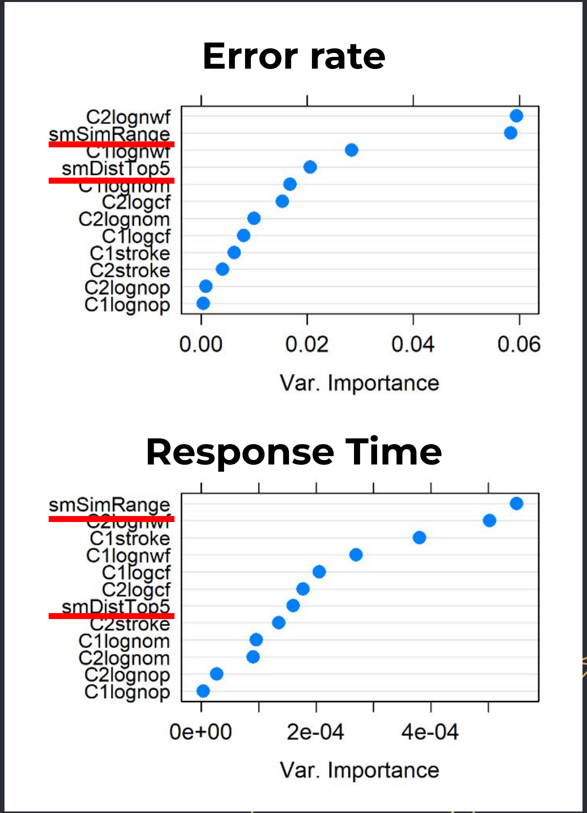
- Given the model learns to predict word embeddings with constituents, what about words that are made-up?
- Pseudowords are often used in psycholinguistic studies. They are "words" that stringing two random characters together, such as "曲車" qǔ chē (literally song-car/bend-car).
- Pseudowords are originally used as "fillers" in psycholinguistic experiments, but studies found responses to these words bear insight into the lexical process. (Yap et al., 2015)

Lexical decision task

- The data we use are the behavioral responses in the lexical decision tasks (LDT).
- Participants sit in front of a computer, and are asked to respond to the stimuli presented on the screen as fast as possible.
- The response is either "word" or "not-a-word". Response time and error rates are computed by each item (pseudoword).
- Here, we used the dataset MELD-SCH, where we took 10K 2-char pseudowords from the dataset (Tsang et al., 2018).

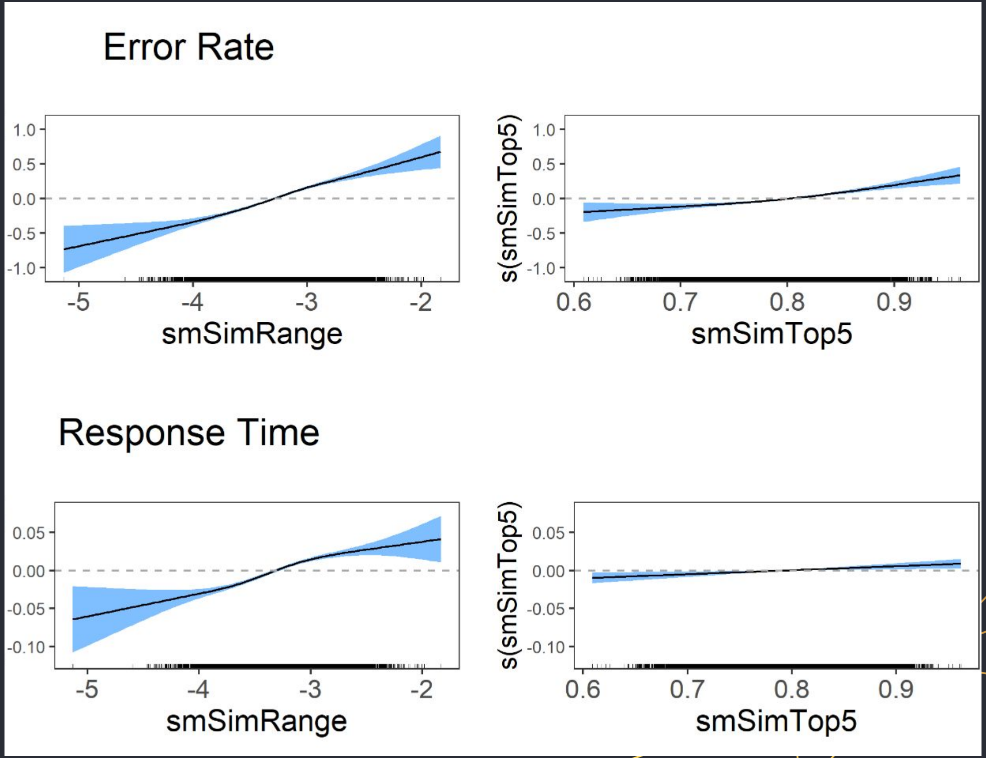
Behavioral data analysis

- Two indices are derived from the Notch model.
- SimRange: how sparsely-populated the pseudoword's location is. Higher the sparser.
- SimTop5: how close the pseudoword is to the real words. Higher the closer.
- Both are important variables when explaining behavioral data.



Statistical Analysis with GAM

- The more sparsely-populated the location is, the higher the error rates and RTs.
- The effects of closeness are weaker, but the closer to real words, the higher the error rates and RTs.



A short discussion of pseudowords

- The model's predicted embeddings of pseudowords help explain the behavioral data.
- Pseudowords are whole-new stimuli to the model, yet we could still derive indices correlated with human behavior.
- One possible account is that pseudowords are still made of characters;
- Chinese characters, while highly polysemous, are also highly systematic. These patterns might be what the model encodes.

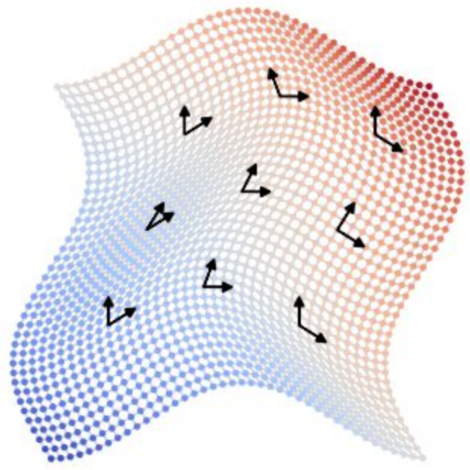
4. Examining characters' meanings with Character Jacobians

Approaches to character meanings

- One possibility is to extract the contextualized embeddings at the output tokens. But they may not be "character-specific."
- Another way to formalize the character meaning in the model is through the Character Jacobians, the arrows in the figure:
- The local transformation of each point in the semantic space

$$\nabla F(X) = \begin{bmatrix} \frac{\partial F_1(X)}{\partial x_1} & \cdots & \frac{\partial F_1(X)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_m(X)}{\partial x_1} & \cdots & \frac{\partial F_m(X)}{\partial x_n} \end{bmatrix}$$

100 x 768



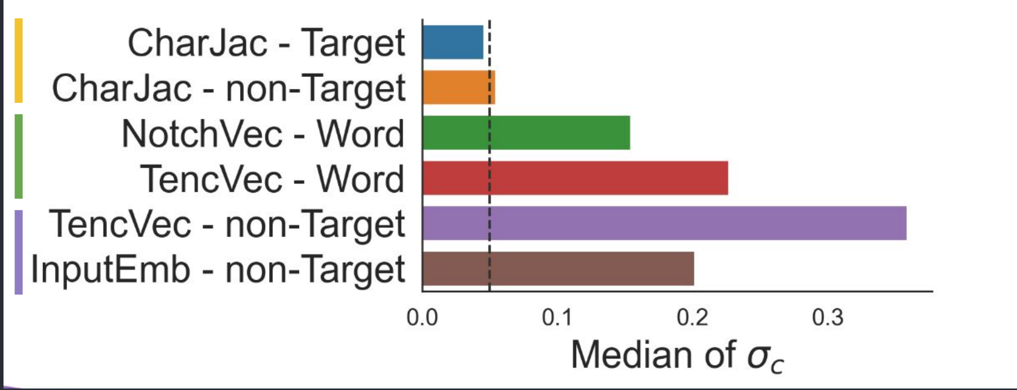
Character Jacobians

- Whether these "black arrows" reflect the meanings of characters.
- Characters with the same meanings should have more similar Jacobians (measured by L1 dist.) than those of different meanings:
  - 土 tǔ, means "*land or clay*"
    in 土石 tǔ shí "*earth and stone*," and 土堤 tǔ tí "*embarkment*"
  - 土 means "*native or local*"
    in 土狗 tǔ gǒu "*native dog*," and 土著 tǔ zhù "*indigenous people*"
- From Common Affixation Database (Academia Sinica, Taiwan), we found 796 unique characters with 1,765 different meanings.

Evaluation by clustering

- We evaluate the similarities within and between the meanings by computing the clustering scores (silhouette scores; Rousseeuw, 1987).
- To better interpret these scores, we randomly permute the data to establish null distributions for each character.
- The score of each character is compared with its own null distribution. A normalized score (σ_c) is computed as the probability of obtaining the values higher than the observed scores given the null distribution.

Results



- **Character Jacobians** perform better than any of the baseline groups. (σ_c, lower the better)
- The 1st set of baseline includes clusterings with **word vectors**: true (TencVec) or predicted (NotchVec). They show how word meanings alone could take us.
- The 2nd set: clusterings with **vectors of the single-char. words** do not perform well: word meanings count, but not through the meaning of varying constituents.

Conclusion

- We propose the Notch model for Chinese compounding.
- It predicts real words' embeddings from their constituents, helps analyze the behavioral data of pseudowords, and models the characters' polysemous behaviors with the Jacobian matrices.
- The methods should in principle apply to other languages. Multilingual application is one of our future works.

Thank you!

Character Jacobian: modeling Chinese character meanings with deep learning models

Yu-Hsiang Tseng, Shu-Kai Hsieh
Graduate Institute of Linguistics
National Taiwan University

@COLING 2022
Gyeongju, Republic of Korea