Vocabulary Associator

How can NLP help for vocabulary acquisition?

David Nukrai, Ameen Ali, Igal Riklin - April 2022

1. Background

1.1 Abstract

Psychology researchers have suggested a novel learning technique that accelerates foriegn language vocabulary acquisition by associating, through visual scenes, the new words with known concepts. Linking the new learned information with known information burns the new concepts faster and stronger both for long-term and short-term memory [10]. The technique is effective and it is especially popular among second language learners for the task of learning new vocabulary fast [7], [8], [9]. Although effective, the technique requires both time and creativity from the learner, which is a major drawback. In this project, we aim to automate the process of creating such associations for the specific task of vocabulary acquisition, by linking the sound and the meaning of the new word with a known word through a meaningful visual scene that is easy to remember. We first suggested a novel algorithm for universal sound matching across different languages and dialects. Then, we use GPT3 [5] to create a meaningful sentence that associates the two words, and finally we use CLIP [3] and Google-photo search engine to find a suitable visual image that illustrates the linking scene. The solution's scheme is described in Figure 1. We finally evaluate our method with 9 different languages and show impressive subjective results.
Our code supports 22 popular languages and it is available at: https://github.com/DavidHuji/vocabulary_associator

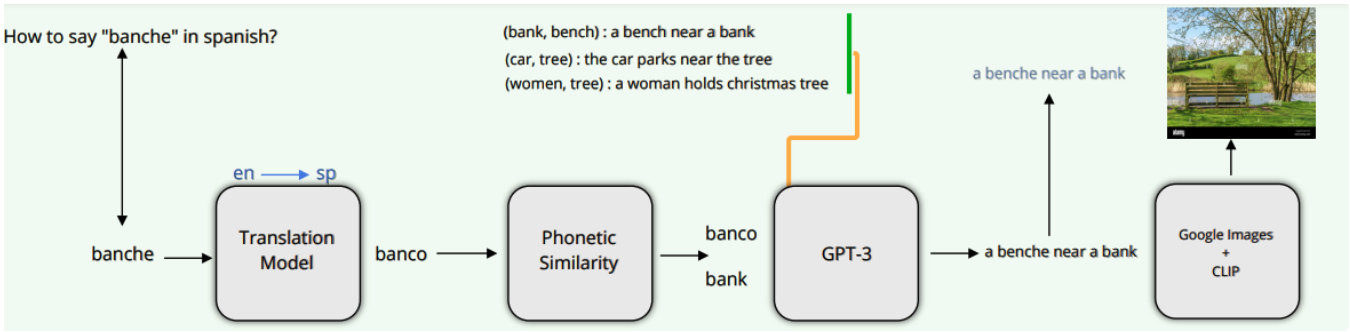

Figure 1

1.2 General description

Our main solution is described in figure 1. Given an input word in forighn language we aim to help learner to remember it by creating a visual association to a known word in the original language. Given a word 'a' from language A, and its translation 'b' to the origin language B we find a word 'c' from language B that sounds like 'a', then we produce a sentence that describes a scene with both 'b' and 'c', and finally we plot an image that illustrates the sentence. Thus, we link the meaning and sound of the new word through a visual scene. It exploits visual,

phenomenological, and associative memory for faster learning which is also fun and not boring and can help to keep focus for longer time.

Our solution works as follows. First we find the sound-similar word, c, by defining a distance function on a phonetic representation of the words. Secondly we use the GPT3 to produce a sentence that describes a scene and contains the two words b and c. Finally we produce an image that helps to remember the sentence and scene by using google photo search and CLIP. We show convening results of our algorithm for the following 9 languages: Hebrew, Arabic, German, French, Spanish, Korean, Russian, Esperanto, and English.

2. Method

As described in figure 1, our method consists of three main components, in this section we will describe each of those components in detail.

2.1 Sound similarity

There are different ways to find similar sounding words. One type of method is SOUNDEX and MATAPHONE. These methods were invented to compare different family names in English, such that homophones will have similar representation. It was used to find the same family name even if it was spelled differently by different people. These methods are generally a set of hardcoded rules. The main problem with these methods is that they are not universal: they work for languages with latin letters, and were built for english pronunciation.

We wanted to find a method that will be universal for all languages and pronunciations. That is why we used IPA (International Phonetic Alphabet). IPA is an alphabetic system of phonetic notation. Different symbols stand for different sounds. There are available pronunciation dictionaries for different languages (the most known one is the English CMU dictionary), where for each word we have its IPA transcript. Then, we need a distance function to compare two words written in IPA.

The PanPhon library's enable mapping IPA segments to 21 Articulatory Feature Vectors, which describe each possible IPA symbol. For each feature the possible values are -1,0,+1.

After converting the IPA symbols to features, the library measures the edit distance between two words. The first one is feature edit distance. It is an edit distance in the feature level. Edit of feature to/from value of 1 cost half from the edit between -1 and 1. Another distance is weighted feature edit distance. The costs of edits change between different features. The weights were found empirically to give better results. We tried to find similar words using the weighted feature edit distance but the results we got were not very good. This measurement tends to miss the main sounds of the word.

In our experiments, we observed that the weighted feature edit distance did not catch the substantial sounds in the word. Thus, we have designed a two stage selection. In the first step we choose a set of words with close main sounds, and only in the second stage we will use the weighted feature edit distance to choose the final candidate within the set.

The first thing we tried was using the SOUNDEX metric. This metric was originally used to find similar family names in the USA census. The main problem is that the soundex works on words in latin transcriptions, and assumes english pronunciation of the word.

The next metric we tried was Dogolpolsky's metric [2]. This metric is the minimum edit distance after collapsing the IPA symbols into several equivalence classes. Phonetic mutations between

the sounds of one class during natural language development had happened more frequently than mutations between sounds that belong to different classes. It was developed in order to research the evolution of languages and language history. However, our goal is different so we took inspiration from the soundex and Dogolpolsky's metric for our own custom metric.
There is a trade off between small and large numbers of equivalence groups. If we divide the alphabet into more equivalence groups, we will find more words within a small distance. However, more of these words with small distance will sound less similar.
We took inspiration from the Dgolpolsky's metric and developed our own SOUNDEX like metric that works on the IPA. Our metric has more equivalence groups, so that more sounds will be regarded differently. We saw empirically that the results are better (subjectively). In one case we did the opposite and combined the s,z group together with the affricated group (ch, dz sounds) that are separate in Dogolpolsky's metric because we thought they are mostly similar. The different groups in SOUNDEX, Dogolpolsky's metric and our metric are described in the attached table.
We also tried to search for pairs of words. For example, the word “mension” is similar to the pair “man” + “shin”. We implemented it by using combinations of pairs of short words; we took all of the pairs of words that are shorter than 5 letters. We only used the top 10k popular words and with the pairs it became 300k words, thus it slowed the tests down significantly since the sorting complexity is higher than linear (nlog(n)), but only seldom improved the results. It can be implemented in a faster way by optimizing the search, and then more pairs of words can be compared against, it is a possible future research path.

2.2 Sentence Generation

We used trained GPT3 to generate the sentences. The main question was which prompt to use. The goal is to generate a short and easy to remember sentence from two given words. We have tried 2 types of prompts: A prompt of instructions and a prompt of example. In the prompt of instruction we give descriptive instruction for the model which sentence to generate. For example “Write a short sentence that contains the words X and Y”. The sentence was indeed generated, but we found that the description could not be detailed. For example when we asked the sentence to be short and not more than 10 words, sometimes the sentence was much longer. We also tried to tell the model to make the sentence “catchy” or “funny”. We did not see that the extra descriptions affected the results.
In the second type of prompt, we give several examples of two words and a short sentence that contains the 2 words. Then, we give the two words we are interested in, the GPT as a result completes it to a new sentence. We encountered a problem that sometimes the GPT continues to generate more content after the sentence. It generates more examples of two words and more sentences. We solved it by adding ‘\n’ between each example in the prompt, and then we have specified ‘\n’ as a stop character for GPT3 which causes the model to stop the autocomplete when this character is produced, and the problem solved.
Another problem we had is that sometimes the sentence the GPT is generating is not using the exact words it was given but rather it uses something that is close in meaning. For example, instead of using the words “nail” it will use the word manicure. Furthermore in some rare cases it did not include one of the words at all. We solved this issue by adding , in the prompt, an

asterisk before and after the two words in the example sentences, then we removed them as post processing. Our final prompt was as follows:

```
"(bank, bench) -> a *bench* near a *bank*
(chair, person) -> a *person* sitting on a *chair*
(car, tree) -> the *car* parks near the *tree
(women, tree) -> a *woman* holds Christmas *tree*
(car, flower) -> there is a *flower* inside the *car*
(wave, grass) -> the wind makes the *grass* move like *waves*
(money, water) -> *water* is *money* in the future
(Google, chair) -> *Google* has a *chair* in its office
({new_word_1}, {new_word_2}) ->”
```

The other parameters of GPT3 we used were: engine="text-davinci-002", temperature=0.7, max_tokens=100, top_p=1, frequency_penalty=0, presence_penalty=0.

2.3 Image Generation

In this component we want to produce a visual image that describes the sentences that were generated at the last component (3.2). We tried to do that by an open source version of DallE but the results were very problematic; many times the image looked not coherent and blurry. Thus, we decided to go with a simple solution of curling google photos search engine. We were surprised to see that the results were actually very good even for non-natural sentences which is because there is an endless amount of images available there. Though we had a problem that about a third of the images were memes or sort of images that only contain a text that is similar to the input sentence. It highly depends on the sentence - if the sentence was too abstract we usually got many memes.

We solved this problem by filtering bad images with CLIP [3]. CLIP is a powerful V&L model that was trained on 400M pairs of images and captions and is able to score similarity between images and text. Clip has been in use in DallE [4] in order to score the similarity between the input sentence and the generated images. Aspiring by the use of Clip in dallE, we chose the image that has the smallest CLIP similarity score to the prompt “a photo of a meme”. The following is an example of an image that was filtered (on the right) for the favor of another image (on the left); it was google photos results of the sentence “you have to go back this way”.

In some rare cases we also encountered other problems with the results of google photos as for example pornographic content which could also be filtered by the same technique of Clip filtering with a suitable prompt. We did not add it to the final code as it was a very rare problem.

3. Experiments and Results

In the attached supplementary, we show output examples of 20 words for each of the following languages: Hebrew, Arabic, German, French, Spanish, Korean, Russian, Esperanto, and English. The words are: ["time", "year", "people", "way", "day", "man", "thing", "woman", "life", "child", "world", "school", "state", "family", "student", "group", "country", "problem", "hand", "part"]. Since the sound matching is sometimes subjective, for each word, we have first shown the top 13 sound matches with the three different distance functions for comparison. Then, for each of the top two sound matches, we present the generated sentence and image.

4. Discussion & Future Work

In this project, we have built a POC that as far as we know is the first solution to the problem of automating the task of creating associations for enhancing vocabulary acquisition. Our solution is based on advanced AI models that were released only last year (i.e. CLIP and GPT3). For the sound matching component, we have developed a unique algorithm that surpasses the current algorithms in the field by adapting the ideas of IPA and Soundex into the unique characteristics of our problem of cross-language sound similarity. Finally, we showed diverse results across 6 different languages in order to evaluate our method on different dialectics.

We think that some improvement may be done in the first component so it will produce better sound matches. We suggest the following options to further research; 1. Create tailor-made Soundex mapping for each pair of languages on top of the IPA rather than having the same mapping for all of the languages. 2. Define sound distance function on top of latent space by first creating a representative latent space by training an autoencoder on the sounds of the words (i.e. mp3 format), and secondly, use L2 distance on this latent space. We have started implementing those methods but we stopped due to the time limitation. It is important to mention that the results of this component are bounded and the optimal solution is bad in some cases because it is not necessarily possible to find a good sound match for every word in a forign language.

The third component of image generation may be improved by text-conditioned generative models as GANs (e.g. DallE [4]) or diffusion models (e.g. Glide [6]). Those models may produce images that better match imaginary sentences of the sentence-generation component, which as can be seen in the results sometimes happens, because realism is not one of the objectives in our sentence-generation component. Furthermore, unrealistic scenes may even enhance the memorization of the scene, thus if we were able to use generative models, we could also work on the sentence generation component so it will produce more creative sentences that are better for memorizing.

This project is a small step towards increasing the human memory bandwidth by outsourcing and automating two of its core tasks; 1. creating associations to known items. 2. using visual

dual coding. Thus, besides the demonstrated application of forign vocabulary acquisition, the spirit of our solution may be applied to many other day to day problems that involve the human memory bandwidth, as for example remembering new names or tasks or remembering new knowledge in general. Such a system will require further development for incorporating complex information into short and ‘easy-to-remember’ visual associations by producing content that associates between the new information and some visualizations of known items (could be applied in diverse media such as video-games or movies, not only images). We hope that our solution will serve as the first baseline for this interesting research line of AI and Psychology.

Appendix 1 - Sound Equivalence Classes

SOUNDEX	Our Metric	Dolgopolsky
V P B F	V P B F W	V P B F
		W
D T	D T	D T
K G S Z	K G	K G H
	S Z CH DG	
		CH DG
	H	
R	R	R L
L	L	
M N	M	M
	N	N
	Y	Y

We use letters instead of IPA symbols for our metric and Dolgopolsky to simplify.
Sounds that are not mentioned (mainly vowels) are ignored.

5. References

[1] David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, Lori Levin (2016). "PanPhon: A Resource for Mapping IPA Segments to Articulatory Feature Vectors." Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 3475–3484, Osaka, Japan, December 11-17 2016.
GitHub:
https://github.com/dmort27/panphon
[2] Dolgopolsky AB (1986) A probabilistic hypothesis concerning the oldest relationships among the language families of northern Eurasia. In: Shevoroshkin VV, Markey TL, eds. Typology, Relationship, and Time: A Collection of Papers on Language Change and Relationship by Soviet Linguists. Ann Arbor (MI): Karoma, pp. 27–50.
[3] CLIP
Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020, 2021.
[4] DallE

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-Shot Text-to-Image Generation. arXiv:2102.12092, 2021.
[5] GPT3
Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. & others (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165, .
[6] GLIDE
A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, “Glide: Towards photorealistic image generation and editing with text-guided diffusion models,” arXiv preprint arXiv:2112.10741, 2021.

[7 - 9] Those links are not papers but examples from regular language learning websites in which the method is suggested - we aim to convince here that the method is in real-world use already; at A, At C, At B

[10] Annette M. B. de Groot - The Learning of Foreign Language Vocabulary