Learning Bilingual Lexicons using the Visual Similarity of Labeled Web Images

Summary

By Ujjwal Kumar Singh(10772)

Paper deals with the problem of word to word translation of a language in another language as for most of the machine translation **parallel data** is unavailable. For the purpose, it uses images to find possible translation of a word. Comparing the different sets of images labeled with words in a two languages (no identical images) to get the efficient translation of word. It could be used to extend the reach of image search engines also.

> " use these explicit, monolingual, image-to-word connections to successfully learn implicit, bilingual, word-to-word translations"

Approach: Used Google's 20 images for each word *candle* (English) and *vela* (Spanish) as their image sets and using color histogram, SIFT keypoints (as visual similarity) and Normalized Edit Distance (as orthographic similarity).
Given words suppose *candle* (English) and *vela* (Spanish), used Google image search to acquire a corresponding set of images, then they extract visual features from these sets by creating color histogram and SIFT keypoints and also used Normalized Edit Distance (as orthographic similarity) where

Color Histogram is partitioned color space containing the count of image pixels(R, G, B) that occur in each partition. They used 4096-dimensional vector space for the experiment.
SIFT keypoints where SIFT stands for Scale Invariant Feature Transform, are features which are immune to scaling, rotation, illumination and distortion and are multidimensional vectors. These features are clustered into K cluster centroids using K-means algorithm (same as k-nn but only we define k nearest to mean rather using distance for it). Signal processing terminology is used to find codeword(cluster centroid in K-dimensional SIFT codebook). Quantization of keypoints is done for a particular image.

"Each dimension in the resulting feature vector corresponds to a codeword; each value is the count of the number of keypoints mapping to that word."

NED: Algorithm to find dynamically number of insertions, deletion and substitutions required to convert one string to another.

$$\textsc{AvgMax}(\mathcal{E},\mathcal{F}) = \frac{1}{|\mathcal{E}|}\sum_{\mathbf{e}\in\mathcal{E}} \max_{\mathbf{f}\in\mathcal{F}}(\text{cosine}(\mathbf{e},\mathbf{f}))$$

$$\textsc{MaxMax}(\mathcal{E},\mathcal{F}) = \max_{\mathbf{e}\in\mathcal{E}} \max_{\mathbf{f}\in\mathcal{F}}(\text{cosine}(\mathbf{e},\mathbf{f}))$$

Above two formulas are used to score the visual similarity of two words using their associated image sets, and then this score for similarity to rank translation pairs for bilingual lexicon induction.

In paper , two ways to create lexicons of physical objects are listed which are later used to perform experiments : precise but low-coverage pattern-based approach and higher-coverage but noisier approach based on distributional similarity with a seed lexicon.

In **pattern-based** lexicon creation rank of words are decided by their conditional probability of co-occurring with the pattern and words are filtered if they occur in corpus < 50% and the first 500 sample is taken for experiment. But there is one problem, matching words with their visual feature.
In **later**, larger list are created that occur in the same context as the seed list for the physical objects. They used the contexual similarity to rank the words(unigrams).

> "We exploit the availability of large corpora in each language to rank a list of unigrams by their contextual similarity with the seeds. Contextual similarity is defined as the cosine similarity between context vectors, where each vector gives the counts of words to the left and right of the target unigram."

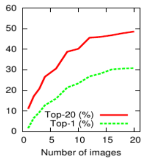
Alternately locally-sensitive hash algorithm can be used to approximate cosine computation and building low dimentional bit signatures and ranking is done by checking their similarity with the ten most similar lexicons and top 20000 is taken for experiment.

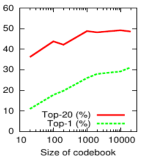
In **Experiment 1** using pattern-based lexicon creation they used the data to calculate AVGMAX, the number of images in each image set (20) and the SIFT codebook dimensionality. MRR is mean reciprocal rank of correct translation and tr(i) is the position of target lexicon with index i .Goal of this experiment is given in image.
Table below shows the result of the experiment which shows %performance and it can be seen AvgMax performs better than MaxMax. AvgMax increases quadratically with the number of images that can be seen in the graph. Increase in codewords results in specific visual representation, not including false positive matches at the cost of more general similarities. They also concluded that using visual similarities along with orthographic similarities results in better performance.

$$\text{MRR} = \frac{1}{500}\sum_{i=1}^{500}\frac{1}{\text{rank}_{\mathcal{E}_i}(\mathcal{F}_{tr(i)})} \text{ (closer to 1 is better).}$$

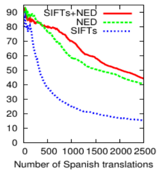
goal is to have $\mathcal{F}_{tr(i)}$ ranked highest, i.e., $\text{rank}_{\mathcal{E}_i}(\mathcal{F}_{tr(i)})$=1.

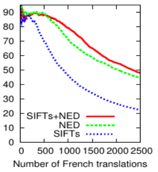
System	MRR	Top-1	Top-5	Top-20
AVGMAX	**36.0**	**31.0**	**40.8**	**48.8**
MAXMAX	31.5	27.0	35.2	42.0





In **Experiment 2 ,**creation of bilingual lexicons using distributional similarities 20,000 words list each having image set of 20 images. Only SIFT and NED are used for this experiment. From below graph it can be easily seen only SIFT doesn't perform well but NED alone improves the performance drastically and SIFT + NED further increase the performance. It also provided the low-frequency noun like fishhook and rosary to be correctly translated which can't be found in many parallel text.





Thus, concluding that bilingual lexicon induction performance can be improved using the labeled images and unrelated language pairs benefits will be further higher and also visual features along with orthographic features provides substantial gain over orthographic features alone.