Towards a Crosslinguistic Identification of Action Concepts. Automatic Clustering of Video Scenes Based on the IMAGACT Multilingual Ontology

Lorenzo Gregori
University of Florence, Italy
lorenzo.gregori@unifi.it

Massimo Moneglia
University of Florence, Italy
massimo.moneglia@unifi.it

Alessandro Panunzi
University of Florence, Italy
alessandro.panunzi@unifi.it

Abstract

This work focuses on the automatic identification of action concepts, performed through machine learning algorithms and applied to a linguistic dataset derived from the IMAGACT ontology of actions. IMAGACT contains a set of 1,010 actions, represented by video scenes, and enriched with linguistic data in several languages. In particular each scene is linked to the full set of verbs that can be used to refer to the depicted action in every considered language. Starting from these data, automatic clustering of scenes has been performed using the linked lexical items as a feature set, following the idea that similar actions can be referred to by a similar group of verbs. Hierarchical agglomerative clustering has been performed with the aim of setting up an evaluation campaign and creating a gold standard of validated clusters. Then, a semi-supervised method based on Affinity Propagation has been trained on these data. An evaluation of clusters coherence has been performed, reporting promising results. An interactive web version of the action map has been also created, to allow users to browse the clusters of videos.

1 Introduction

IMAGACT (Moneglia et al., 2014) is a multilingual ontology of action consisting in a fine-grained categorization of action concepts, each represented by prototypes in the form of recorded videos and 3D animations. Action concepts have been identified though the annotation of Italian and English corpora of spontaneous speech (Moneglia et al., 2012). Starting from the occurrences of verbs referring to physical actions, the set of different Action concepts to which each verb can extend has been retrieved. Conversely, the set of verbs which are able to refer to the same Action Concept has also been identified (Panunzi et al., 2018). Reconciling the annotation derived from the two language corpora, a set of 1,010 scenes has been generated, each one representing a prototype for an Action Concept. This set of scenes encompass the actions commonly referred to in everyday language usage.

The insertion of new languages beyond Italian and English has been obtained exploiting the Competence Based Extension (CBE) technique (Brown et al., 2014). Using a method of ostensive definitions, informants of different origins have been asked to list all the verbs in their mother-tongue that can refer to the action depicted in a given prototypical scene. At present, IMAGACT contains 16 fully mapped languages, and several others are underway. The visual prototypes are therefore relevant for cross-linguistic reference.

However, in IMAGACT the relation among prototypes remains undefined. They can only be linked by linguistic correlations expressed by action verbs referring to them from the perspective of a single language (Table 1). How similar actions can be grouped into more general Action Types and how near or far Action Types are in the topological space of actions is not defined, and should be in principle independent from the perspective of a single language. For instance, in Table 1 actions (a), (b) and (c), may be considered similar from the perspective of English and Italian, since all are in the extension of one general verb (*put* and *mettere*, respectively), even though different equivalent verbs (*hang*, *spread*, and *lay*) mark their differential. However, this cannot be the case from the perspective of Japanese which does not have a single general verb extending to these 3 actions. Moreover, even from the perspective of English, we do not know whether (c) is more similar to (a) and (b) or instead to (d), which shares with (c) the equivalence with *spread*, while in Italian (c) and (d) have no equivalence.

The basic strategy developed in this study is to exploit the full set of cross-linguistic correlations

		(a)	(b)	(c)	(d)
EN	*put*	✗	✗	✗	
EN	*lay*		✗	✗	
EN	*hang*	✗			
EN	*spread*			✗	✗
IT	*mettere*	✗	✗	✗	
IT	*spalmare*			✗	
IT	*appendere*	✗			
IT	*sparpagliare*				✗
JP	掛ける *(kekeru)*	✗			
JP	付ける *(tsukeru)*			✗	
JP	置く *(oku)*		✗		
JP	散らばる *(chirabaru)*				✗

Table 1: Verb-to-scene reference example.

of each prototype in IMAGACT to generate an ontological space where actions are clustered through machine learning algorithms. This should avoid bias from a single monolingual-centric approach, but will be consistent with the idea that similar actions can be referred to by a similar group of verbs.

This paper follows a previous promising work (Gregori et al., 2019) based on the Affinity Propagation algorithm which, despite the admirable results achieved, remained unsupervised. The evaluation of the resulting clusters is not trivial, since we need to compare one speaker's conceptual representation with the average representation resulting from summing lexical information from multiple languages. Our present goal is to set up a strategy for developing a gold standard of validated clusters of action prototypes. In Section 2 the dataset settled to this end will be illustrated. In Section 3 we will present the strategy of using a HAC for grounding an evaluation campaign, and we will outline how similarity judgements have been obtained from informants in a crowdsourcing infrastructure. In Section 4 the results of a semi-supervised method based on Affinity Propagation will be presented and evaluated in terms of cluster coherence. We will finally show the action map, which allows users to browse the clusters of videos based on the algorithm results.

2 Dataset creation

Our dataset is created from the IMAGACT database, using the same technique as the previous work (Gregori et al., 2019), but with significant updates to CBE data (action videos and referring verbs). The raw dataset is a binary matrix $C_{1010\times10572}$ with one row per video and one column per verb (belonging to 14 languages). Matrix values are the assignments of verbs to videos made by native speakers within the CBE annotation task:

$$C_{i,j} = \begin{cases} 1 & \text{if verb } j \text{ refers to action } i \\ 0 & \text{else} \end{cases}$$

Matrix C encodes the inter-linguistic lexical representation of each video.

Table 2 shows the number of verbs assigned by the CBE annotators for each language. It is important to notice that the task has been performed on the whole set of 1,010 scenes for each language, and that the differences between the number of verbs depend on various linguistic factors. Some examples of verb-rich languages are: (a) Polish and Serbian, in which perfective/imperfective forms are lemmatized as different entries; (b) German, which has particle-verb compositionality; (c) Spanish and Portuguese, for which verbs belong to both American and European varieties.

An approximated matrix C' is created from C, by using *Singular Value Decomposition* (SVD) and truncating to 300 dimensions. Dimensionality reduction allowed us to obtain a fixed-size feature

Language	Verbs
Arabic (Syria)	571
Danish	646
English	673
French	669
German	990
Greek	631
Hindi	449
Italian	668
Japanese	736
Polish	1,192
Portuguese	776
Serbian	1,081
Spanish	735
Swedish	755
TOTAL	**10,572**

Table 2: Number of verbs per language.

space, and an approximate matrix that smooths over language-specific semantic differences. C' matrix has been used as a working dataset for the current clustering task.

3 Hierarchical Agglomerative Clustering

The main issues in clustering the IMAGACT dataset are:

- unknown number of clusters: in the previous experiment (Gregori et al., 2019), 178 clusters were automatically created, but here data and method have changed, so we cannot impose a set number of clusters;
- high potential variability in cluster size: given that in every language there are general verbs (that can refer to many videos) and specific ones (that can refer to just one or few videos), we can expect similar variability in cluster size.

In this scenario, the Hierarchical Agglomerative Clustering (henceforth, HAC) algorithm is a suitable choice: in fact, differently from other algorithms (e.g. K-means), it does not require the programmer to preset the desired number of clusters, and performs an unbiased grouping based only on elements' similarity. Elements' closeness has been computed through the cosine similarity measure. Given a dataset with N elements, HAC performs the following steps:

1. The algorithm is initialized by creating one cluster per element, resulting in N clusters of one element each;
2. At each step, the two nearest clusters are merged; cluster distance is measured according to a specified metric;
3. In the last step all the elements are merged into one cluster.

A full run of HAC produces a wide set of possible clustering outputs, and a stopping criterion is required to obtain a reasonable number of clusters. To this aim, an automatic analysis and a manual evaluation have been performed on the outputs.

3.1 Automatic evaluation with the silhouette coefficient

The silhouette score (Rousseeuw, 1987) measures tightness and separation of clusters resulting from any method. This metric returns a value between -1 and +1, with the following meaning:

- A score near 1 highlights strong tightness of intra-cluster elements and a good inter-cluster separation;
- A score near 0 represents highly overlapping clusters;
- A score near -1 indicates that the elements are assigned to the wrong clusters;

Silhouette coefficient SC (Kaufman and Rousseeuw, 1990) extends the silhouette score to an entire dataset. Automatic clusters analysis on the current dataset has been performed by computing the SC for every possible number of clusters that can be obtained at each step of the hierarchical clustering algorithm. According to the algorithm, the number of clusters obtained ranges from 1 to 1010 (that is the number of video instances). Figure 1 shows the SC value for each number of clusters.

The plot clearly shows that the highest values of SC are obtained with a number of clusters that is in the middle, while SC decreases rapidly if we move towards the minimum or maximum number of clusters. In particular we observe:

- $SC > 0.30$ with a number of clusters between 290 and 770;
- $SC > 0.40$ with a number of clusters between 415 and 589.

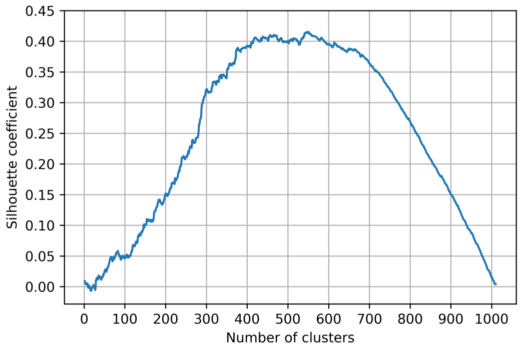

Figure 1: Silhouette score for each number of clusters.

3.2 Manual evaluation with crowdsourced similarity judgments

Results obtained with the silhouette coefficient are easy to interpret and allow the creation of clusters that are optimized for internal tightness and external separation. What we still don't know is whether the clusters created actually make any sense for a human. For this reason, we have set up a procedure to collect human judgments and validate the clustering results. A set of surveys has been designed in order to determine to what extent clusters can be grown preserving a perceived coherence.

Informants have been forewarned that the evaluation of similarity among events is by definition a vague task. It may regard features of different value, such as non-essential attributes like the presence of similar objects, the overall circumstance, the mood of the performance or more abstract similarities such as the goal of the action. The informant has been explicitly asked to disregard superficial similarities and to judge whether or not *what happens* in the events under consideration is similar.

The procedure does not show the informant the full set of scenes in a single viewing. This kind of evaluation would be costly in terms of time, attention and reasoning capacity, since it would require the simultaneous comparison of many possible similarities in a large set of different scenes. To make the evaluation procedure easier and more reliable we implemented an incremental test which allows the informant to build up his interpretation little by little, starting from a simple similarity judgment between two scenes. The procedure runs as follows.

A set of 283 surveys has been created, each one containing the chain of scenes obtained from the clustering output. From the 283 surveys, the first 100 have been published (∼35%), and 6 raters have evaluated them. Finally, we have obtained 6 cutting thresholds for each chain of videos. LimeSurvey[1], an open-source online survey tool, has been used to set up and submit the evaluation task.

The scenes belonging to each chain are presented one by one to the human evaluator in order to mimic the algorithm behaviour. The scenes are ordered from the closest to the farthest, according to the hierarchical clustering output.

Given a first pair of scenes produced by the system, the informant is asked to choose among four alternatives on a similarity scale, ranging from very similar to different. *What happens* can be judged:

1. similar (can be gathered in a group of events of the same kind);
2. quite similar (there are differences, but it can be still considered an event of the same kind, although in some sense peripheral);
3. quite different (some similarity with the group can be seen, but it should be kept distinct);
4. different (no meaningful similarities).

A video with a rating of 1 or 2 fits well in the cluster (and is colored green during the survey); a video with 3 or 4 should be put outside the cluster (and is colored red during the survey).

If the two scenes are considered events of the same kind, the informant goes on with the test judging the similarity of other scenes (one by one) with respect to what happens in the first pair. When three subsequent scenes are judged different or quite different and are marked in red the test is interrupted.

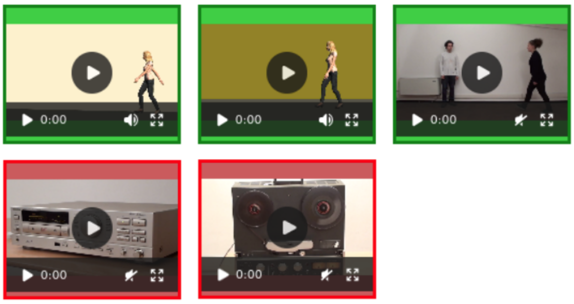

Figure 2: Example of an evaluation survey.

For instance, in Figure 2 the three events in the first row (in green) have been judged similar, while the actions represented in the last two videos (in red) have been excluded from the cluster.

[1] https://www.limesurvey.org/

3.3 Results

A single threshold value has been computed for each video chain, by averaging the ratings. A video is considered to fit into the cluster if the mean of its ratings is lower or equal to 2.5. Table 3 shows an example of rating results for a survey with 6 videos (Q0 to Q5), evaluated by 4 annotators (column Q00 reports the annotator ID). The mean value of each column has also been reported to highlight the cutting threshold: the first 4 videos have a mean rating below 2.5, so they are similar enough to be clustered together, while the last 2 videos have a mean higher than 2.5, so are considered outside the cluster.

Q00	**Q0**	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**
cc	1	3	3	3	3	3
st	1	2	3	2	3	3
vs	2	1	1	1	4	4
lg	1	3	3	3	3	3
mean	**1.25**	**2.25**	**2.5**	**2.25**	**3.25**	**3.25**

Table 3: An example survey result: a chain of 6 videos evaluated by 4 annotators.

In order to convert the results reported in the example above (Table 3) into a stopping criterion for the clustering algorithm, we can determine at which steps of the algorithm a cluster contains exactly 4 videos. All that is needed to obtain this criterion is to find the step at which the 4th video is added to that cluster, and the step at which the 5th video is added to that cluster: all the steps in between are optimal stopping points for this example.

The range of optimal stopping points has been computed for each of the 100 annotated surveys. The optimal number of clusters is trivially computed with the inversion of the previous list[2]. Results are displayed in Fig. 3.

Similarly to the silhouette values, the optimal number of clusters is in the middle, and the correctness rate quickly decreases if we move towards the minimum or the maximum number of clusters. In particular we observe:

- Correct clusters > 35 with a number of clusters between 299 and 320, and between 344 and 870;
- Correct clusters > 45 with a number of clusters between 543 and 552, and between 568 and 732.

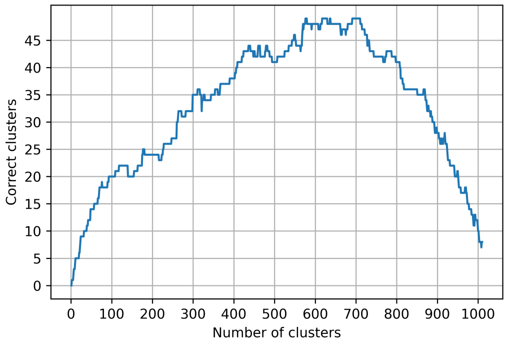

Figure 3: Graph of the number of best-fitting chains for any number of clusters.

3.4 Optimal number of clusters estimation

Figure 4 summarizes the results:

- light green stripes show the ranges of clusters number, reporting a relevant number of correct clusters (more than 35), according to human evaluation;
- dark green stripes show the ranges of clusters number, reporting the highest number of correct clusters (more than 45), according to human evaluation;
- light purple stripes show the ranges of clusters number, reporting a good silhouette coefficient ($SC > 0.30$);
- dark purple stripes show the ranges of clusters number, reporting the optimum silhouette coefficient ($SC > 0.40$).

Considering these results, along with the obvious propensity to keep the number of clusters as small as possible, it is possible to derive the following parameter estimation:

- $\kappa = 543$ is the optimal number of clusters, reporting the highest SC score and the highest number of correct manual evaluations;
- $\kappa = 299$ is still a good number of clusters (high SC score and relevant percentage of correctness), and offers a more compact representation.

Finally, one of the two κ values can be chosen as threshold, depending on the purposes: precision maximization, or reduced number of clusters.

[2] Note that at step 1 there are N clusters, and at step N there is 1 cluster.

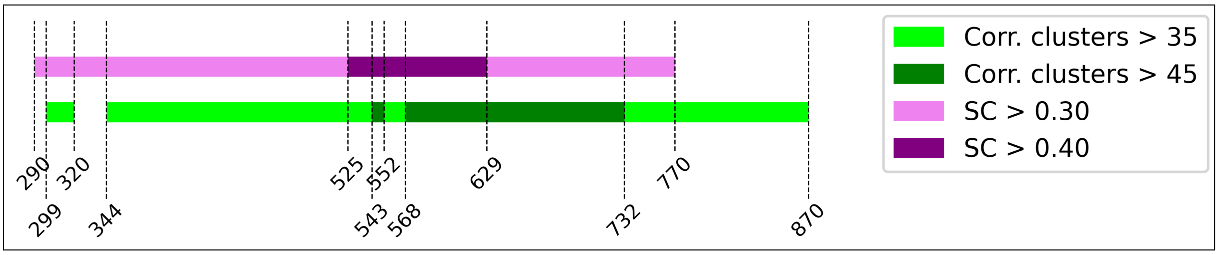

Figure 4: Good and optimal number of clusters, according to silhouette coefficient and manual annotation.

3.5 Agreement

Inter-annotator agreement has been measured with Fleiss' Kappa (Fleiss, 1971) on annotated data. In particular, the agreement on the number of clusters that are part of each chain was measured. Fleiss' Kappa value among 6 annotators was computed for each of the 100 surveys: excluding a few outliers, agreement ranges between 0.25 and 0.5, with a prevalence around 0.3 (Figure 5).

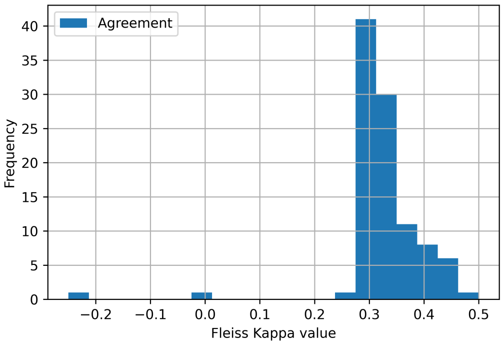

Figure 5: Inter-annotator agreement measured with Fleiss' Kappa on 6 annotators.

In addition to the Fleiss' Kappa metric, Figure 6 shows a bar plot with the number of surveys with different deviations from the optimal number of clusters for each annotator. For example, the blue bars (deviation = 0) show the number of surveys for which the number of clusters belonging to the chain is equal to the optimal number of clusters (estimated by averaging raters' judgments).

This figure gives a broader picture of the inter-rater agreement, showing that, despite the exact number of clusters in the chains not being widely shared (average Fleiss' Kappa is about 0.3), the size of the clusters is near the optimum (with a deviation of 0, 1 or 2 scenes) for most of the ratings.

Nonetheless, even after tuning, the accuracy of hierarchical clustering is still low: even with the optimal number of clusters ($\kappa = 543$), more than 50% of the evaluated clusters does not match with our gold standard derived from manual annotation. This result suggests that HAC is not a good choice to segment our dataset.

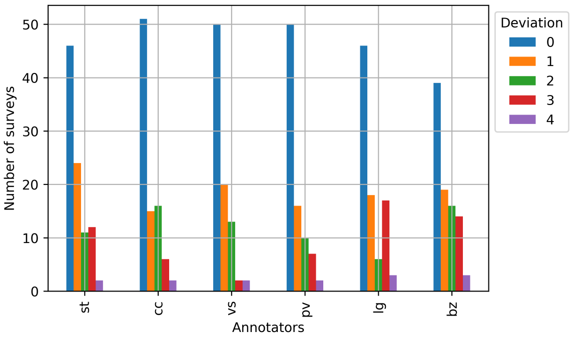

Figure 6: Number of surveys with different deviations from the optimal number of clusters for each annotator.

4 Informed Affinity Propagation clustering

Despite these negative results, hierarchical clustering provides a way to set up an evaluation task, and allowed us to create a gold standard, exploitable for training a semi-supervised clustering algorithm.

An attempt to perform semi-supervised clustering has been made through the introduction of a bias into the Affinity Propagation (henceforth, AP) algorithm. AP can be tuned through a vector of *preferences*, which adjusts the probability of each data point being in the center of a cluster: if an element has a higher preference value, it has a higher probability of being near the centroid of a cluster. Starting from our gold standard of 100 clusters, we selected the scene of each cluster that is nearest the cluster centroid. Then, we increased the preference value of these scenes and ran the AP. The introduction of this bias is expected to enforce the algorithm to perform a clustering that is closer to our gold standard.

Figure 7 shows the clustering accuracy for different values of the preference parameter[3]. Accuracy

[3] Preference values range between the minimum and the

is measured by comparing the automatic clustering with the provided gold standard; we used Adjusted Rand Index as our comparison metric.

The black line reports the accuracy of AP with a uniform preference value for each element. Then, we introduced a positive bias, by increasing the preference on the 100 scenes that are near the centroids of validated clusters (green line). Finally, we performed a final test, penalizing the preference of these 100 scenes (red line), to better appreciate the impact of our bias.

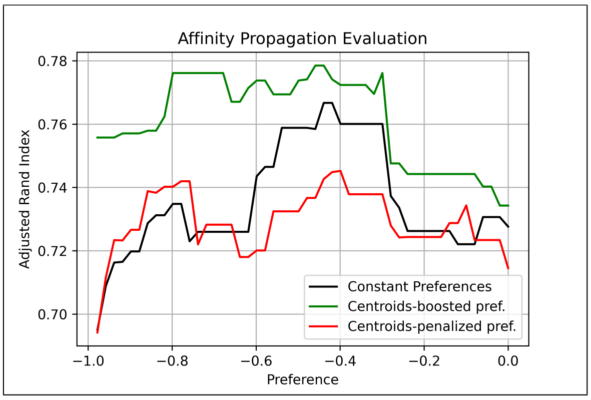

Figure 7: Cluster correctness with different values of *preference* parameter in AP.

Results clearly shows that the introduction of a positive bias increases the matching with our gold standard, while a negative bias reduces the accuracy with respect to a default run with constant preferences. Moreover, we estimated two constant values of preference for unbiased scenes[4]:

- $pref = -0.44$: the algorithm generates 182 clusters; the Adjusted Rand Index is maximum;
- $pref = -0.82$: the algorithm generates 155 clusters; the Adjusted Rand Index is near the maximum.

4.1 Voronoi diagram

In order to obtain a visual representation of clusters, the following actions have been performed:

- centroids computation: centroids have been computed for each group of scene vectors belonging to a cluster, by averaging their values; one 300-dimensional vector per cluster has been derived;
- 2D representation of centroids: the t-SNE algorithm has been applied to the centroids to reduce the 300 dimensions to 2 dimensions, thus obtaining one point per cluster;
- creation of a Voronoi diagram: the points have been projected in a 2D space, and a Voronoi diagram has been created, to display clusters as spatial areas; one polygon per cluster has been generated (Fig. 8).

This representation allows us to automatically draw a picture of the space of events. Since the proposed clustering involves videos, an HTML representation of the Voronoi diagram has been generated. This has made it possible to create an interactive map, where a user can browse the diagram and play the video elements[5]. Figure 8 shows a screenshot of the map, which can be accessed[6] online.

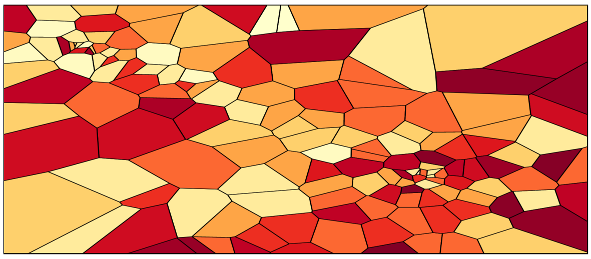

Figure 8: Voronoi diagram of clusters.

4.2 Clusters coherence evaluation

Clusters evaluation aims at verifying, firstly, whether the clusters obtained are meaningful for a human. Then, we can also verify whether the benefit obtained by introducing a bias is local or global, i.e. if the clusters with biased scenes are *as good as* the others or not. Finally, we are interested in verifying whether the error of the two runs of the algorithm (with different preference values) is similar or not.

Rather than a strict similarity among scenes, the evaluation of the clusters produced by AP judges their overall internal coherence. For each evaluated cluster, the informant is asked to discard the scenes which should not be grouped in the same cluster, if any.

For instance, the cluster in figure 9 brings together a set of events in which the agent places

median of the dataset, as suggested by (Frey and Dueck, 2007).

[4] Biased scenes have the same preference value plus 1.0; the results are nearly the same with higher or lower values.

[5] In the diagram, the darker areas correspond to the clusters with the higher number of scenes.

[6] http://lablita.it/app/imclust/voronoi2.html

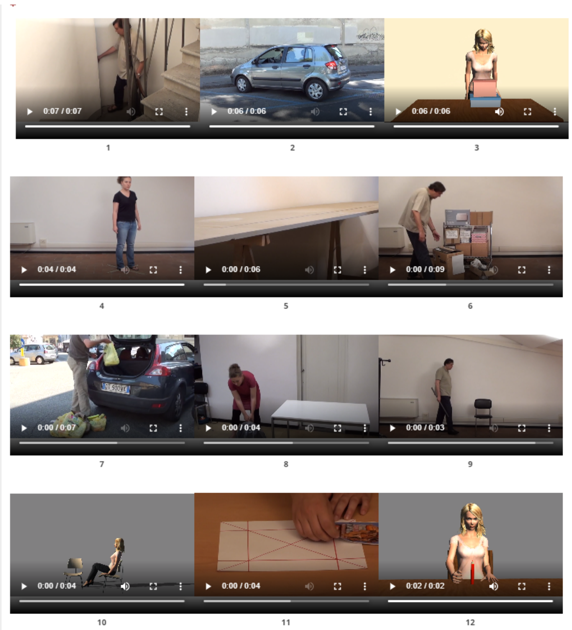

Figure 9: An example of a generated cluster.

an object somewhere in different manners, but the cluster also collects the parking of a car and other scenes in which the subject places himself in a certain position. The informant found it odd to have these last events in the cluster and decided that they were not coherent (even though the general English verb *to put* can be applied to all scenes, like many other verbs at a cross-linguistic level; that is probably the reason why the algorithm formed the cluster).

Clusters coherence evaluation has been performed by 5 raters on 120 clusters: 60 clusters belong to the output of AP with $pref = -0.44$, while the other 60 are generated with $pref = -0.82$. In both of these sets, 30 clusters contain a biased scene and 30 clusters are unbiased.

For each cluster, we counted the relative number of scenes that are evaluated as not coherent by each annotator, i.e. the percentage of *wrong* scenes in each cluster. Then, we computed the mean of these values for the 5 annotators. Finally, the overall mean has been derived for each of following subset of 30 clusters:

- Average percentage of incoherent scenes in biased clusters created through AP with $pref = -0.44$;
- Average percentage of incoherent scenes in unbiased clusters created through AP with $pref = -0.44$;
- Average percentage of incoherent scenes in biased clusters created through AP with $pref = -0.82$;
- Average percentage of incoherent scenes in unbiased clusters created through AP with $pref = -0.82$;

Results are summarized in Table 4. The table highlights that, on average, raters found 15% of incoherent scenes in the biased clusters (i.e. the clusters containing a scene with increased preference value), independently of the global preference value of AP. Conversely, the error measured on the clusters with constant preference is higher and dependent on the global preference value: 18% and 21% of incoherent scenes.

	$pref = -0.44$	$pref = -0.82$
Err. on bias	0.15	0.15
Err. on unbias	0.18	0.21

Table 4: Relative error measured on the perceived coherence of the clusters generated by AP.

We can derive that introducing a bias on the basis of a gold standard leads to an accuracy improvement on AP, and the error can be estimated as ∼15% of incoherent scenes in both the clusterings (with 155 and 182 clusters). In addition to this, the error on unbiased clusters is higher and not stable in the two algorithm runs, meaning that a local bias does not bring a global benefit.

5 Conclusions

We have presented our work on the automatic identification of action concepts exploiting lexical data belonging to 14 languages, stored in the IMAGACT multilingual ontology of action. The automatic clustering procedure consisted in two main steps. First we applied the HAC method obtaining negative results. Nevertheless, the evaluation campaign was used to bootstrap a gold standard of validated clusters, on which we trained a semi-supervised method based on Affinity Propagation. The evaluation of the cluster coherence of this second method gave promising results, which must be further refined in the forthcoming steps of our research.

References

Susan Brown, Gloria Gagliardi, and Massimo Moneglia. 2014. Imagact4all mapping spanish varieties onto a corpus-based ontology of action. *CHIMERA. Romance Corpora and Linguistic Studies*, 1:91–135.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Brendan J Frey and Delbert Dueck. 2007. Clustering by passing messages between data points. *science*, 315(5814):972–976.

Lorenzo Gregori, Rossella Varvara, and Andrea Amelio Ravelli. 2019. Action type induction from multilingual lexical features. *Procesamiento del Lenguaje Natural*, 63:85–92.

Leonard Kaufman and Peter J Rousseeuw. 1990. Finding groups in data; an introduction to cluster analysis. Technical report, J. Wiley.

Massimo Moneglia, Susan Brown, Francesca Frontini, Gloria Gagliardi, Fahad Khan, Monica Monachini, and Alessandro Panunzi. 2014. The imagact visual ontology. an extendable multilingual infrastructure for the representation of lexical encoding of action. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Massimo Moneglia, Francesca Frontini, Gloria Gagliardi, Irene Russo, Alessandro Panunzi, and Monica Monachini. 2012. Imagact: deriving an action ontology from spoken corpora. *Proceedings of the Eighth Joint ACL-ISO Workshop on Interoperable Semantic Annotation (isa-8)*, pages 42–47.

Alessandro Panunzi, Massimo Moneglia, and Lorenzo Gregori. 2018. Action identification and local equivalence of action verbs: the annotation framework of the imagact ontology. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.