Language-Action Tools for Cognitive Artificial Agents: Papers from the 2011 AAAI Workshop (WS-11-14)

Language Models for Semantic Extraction and Filtering in Video Action Recognition*

Evelyne Tzoukermann[1], Jan Neumann [2], Jana Kosecka[3], Cornelia Fermuller[4], Ian Perera[5], Frank Ferraro[6], Ben Sapp[5], Rizwan Chaudhry[7] and Gautam Singh[3]

[1]The MITRE Corporation, McLean, VA; [2]Comcast, Washington, DC; [3]George Mason University, Fairfax, VA; [4]University of Maryland, College Park, MD; [5]University of Pennsylvania, Philadelphia, PA; [6]University of Rochester, Rochester, NY ; [7]Johns Hopkins University

tzoukermann@mitre.org;neumann@cable.comcast.com; kosecka@cs.gmu.edu; fer@umiacs.umd.edu; eperera@seas.upenn.edu; fferraro@u.rochester.edu; bensapp@cis.upenn.edu; rizwanch@cis.jhu.edu; gsinghc@cs.gmu.edu

Abstract

The paper addresses the following issues: (a) how to represent semantic information from natural language so that a vision model can utilize it? (b) how to extract the salient textual information relevant to vision? For a given domain, we present a new model of semantic extraction that takes into account word relatedness as well as word disambiguation in order to apply to a vision model. We automatically process the text transcripts and perform syntactic analysis to extract dependency relations. We then perform semantic extraction on the output to filter semantic entities related to actions. The resulting data are used to populate a matrix of co-occurrences utilized by the vision processing modules. Results show that explicitly modeling the co-occurrence of actions and tools significantly improved performance.

Introduction *[1]*

We present the language models of an end-to-end system capable of automatically annotating real-word broadcast videos containing actions and objects. As shown in Figure 1, the input to the system is a set of broadcast videos with transcripts. In the preprocessing stage, videos are segmented and clustered in shot boundary and face recognition segments groups. Then, for each shot, local models of detection and segmentation are applied in order to localize objects and hands. Visual processes are then combined with language models. The global model integrates multiple visual features and provides temporal information along with the action type. Results show the impact of language models in the overall vision system.

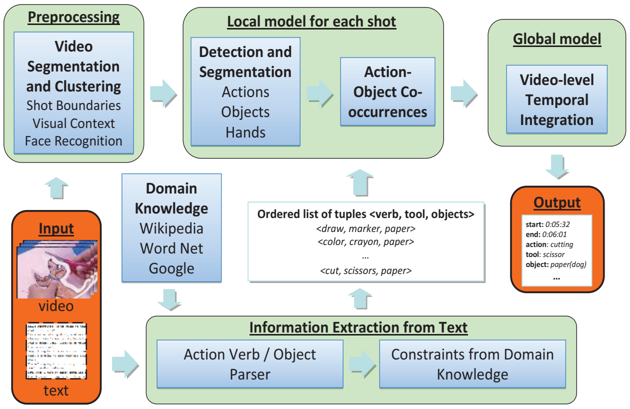

Figure 1. System Overview

We address the natural language aspects of a temporal probabilistic model. Applying natural language processing techniques to the textual descriptions corresponding to videos provides both semantic and temporal information about the actions in the videos. More specifically, we automatically process the text transcripts and perform syntactic analysis to extract dependency relations and perform semantic extraction on the output to filter semantic entities related to actions. We then use existing ontologies, such as Wikipedia, WordNet, ConceptNet, as well as the web to determine the semantic relatedness of verbs and objects and objects and instruments. The resulting data are used to populate a matrix of co-occurrences utilized by the vision processing modules. The textual descriptions form

*The work presented here was performed during the 2010 CLSP summer workshop at the Johns Hopkins University.

the contextual priors along with multiple video features are used by the vision module to learn a probabilistic temporal model of a video clip.

The following section presents the front end language processes where syntactic parsing and semantic filtering are applied. After, we show different models of semantic extraction from knowledge bases. We present the language work in the existing related literature and demonstrate the integration of language processes with vision modules. We conclude in presenting ideas for future work.

Parsing and Extracting textual Information

This section first presents the dataset and is followed by a discussion of the syntactic parsing and semantic extraction performed on the texts.

Dataset: The dataset consists of 27 PBS Sprout Arts and Craft shows for which both the video and the transcript are available. There are usually 40 sentences per show. The transcripts are generally of very high quality and contain few mistakes. They are also very structured in that each show is a dialog between the same two characters – the host Nina and her co-host Star. Since these shows consist of teaching how to construct a piece of arts and craft, we might also assume that the set of instructions renders the shows semantically dense, i.e. containing a significant number of words that are useful to performing or detecting a given task. Unfortunately, since the show is a dialog aimed at being broadcast, the texts are semantically sparse. This is true of any domain of instructions, e.g. home improvement shows. The shows are more narrative in nature rather than imperative. For instance, a wrapping action (something our system should detect) can be given by the conditional clause “if you just wrap the tape around [the yarn], it'll make it so much easier to finish this”. This has the consequence of not guaranteeing simple sentence structure. Although some instructions are very clearly and simply stated (such as “cut the paper”), others are hidden within the narrative, as demonstrated above. Further, since the shows are in the Arts and crafts domain, determining the richness of the vocabulary is difficult, as nearly anything can be used in a craft: a rock, a coffee filter, an egg shell, a bottle cap, etc. Although the use of synonyms may be limited, the set of possible objects used is extremely large. Thus, this domain of application is even harder than on-line learning workshops which tend to be more limited in vocabulary.

The shows were annotated by humans who provided action verbs, objects, and instruments, along with time stamps.

In order to aid the recognition of actions in videos, we used three specific natural language techniques: (1) syntactic parsing, (2) extraction of syntactic and semantic entities, and (3) extraction of semantic information from domain knowledge.

Extraction of Domain Knowledge

We used domain knowledge to aid in object and action recognition by using textual information about the videos as seed data to extract semantic information from larger knowledge sources, including the web. This process results in semantic background in the form of a co-occurrence matrix, which gives a likelihood of a tool and action occurring at the same time in a video.

Co-occurrence Matrices

To create co-occurrence matrices, we experimented with three different sources of knowledge, (a) Wikipedia along with WordNet, (b) ConceptNet, and (c) the web using Yahoo search results. Sources (a) and (c) are used in conjunction with each other in the global model which integrates vision and text features.

Wikipedia Matrix

Along with Wikipedia to extract the relationships, we also use WordNet, a lexical database for English that categorizes words into semantic categories (Fellbaum, 1998). It is used as a dictionary, or to find relationships between related words. Our method for finding action-tool relationships in Wikipedia consists of the following: (a) look up the page corresponding to a particular action, (b) find the nouns, and (c) check WordNet to see which nouns are tools. This is somewhat similar to the approaches used by Ruiz-Casado et al. (2005) where they extract semantic relationships to extend existing ontologies.

We improved precision at a slight cost to recall by checking nouns only within links and captions in the Wikipedia page. Captions also provide useful information, because we are ultimately interested in the relationship between textual semantics and vision information. For example, the page for “gluing” gives a caption, “Nitrocellulose adhesive outside a “tube” but “tube” does not appear anywhere else in the document, even though it directly correlates to a shape that we would expect to see in video. We used WordNet’s relationship properties to check the parsed words which are given for each synset. To find whether a word could be a tool, we retrieve the hypernym paths of each synset containing that word. The hypernym paths provide all of the lexical categories that the synset belongs to – from this list, we can find whether the object belongs to the “implement” or “tool” category. One drawback of using Wikipedia is that words are not sense-tagged. We must therefore look for nouns that have any possible sense that is a tool, regardless of whether it is used in that context.

This approach ensures high recall, but low precision. While there are some sense disambiguation tools available online, those we evaluated on Wikipedia pages were not accurate enough to use.

	color	cut	draw	glue	paint	place
brush	0	0	1	0	1	0
writing implement	1	0	1	0	0	0
glue	0	0	0	1	0	0
scissors	0	1	0	0	0	0

Table 1: Wikipedia Matrix

Results: As shown in Table 1, the results from Wikipedia were promising in that they matched both intuition and the actual co-occurrences of our training data. For this research, we used a list of tools and actions that we knew would appear in the test data, and restricted the co-occurrence matrix to include only those tools and actions. However, our results could also be used if we did not know the tools (or actions) that would appear in video. For example, we used online image databases such as image.net or Google Images to retrieve images for a particular tool to train on. An alternative option would be to find physical characteristics of these objects from domain knowledge on the web, and look for these characteristics in the video.

ConceptNet

We explored ConceptNet as another knowledge base to extract action-tool representations (Liu 2004). ConceptNet is a common-sense semantic network where semantic relationships are added to the database by any web user that has registered for the Common Sense Initiative website. This database includes relationships such as *PropertyOf*, *IsA*, and *UsedFor* to define words in terms of semantic relationships with other words. Within ConceptNet, we captured the relationships *UsedFor* or *CapableOf* in order to find the actions a tool is used for. The same relationships were used to find the tools used for a particular action and we searched for all occurrences of the relationship with the action as the second argument. Since ConceptNet relies on contributions from users of the Common Sense Initiative and not supplied by experts, there are many gaps in the knowledge base. While ConceptNet does use algorithms for inducing relationships, there were many relationships we expected but did not find. There were also some errors in the database, such as the tool "saw" being stemmed to form the word "see".

	color	cut	draw	glue	paint	place
brush	0	0	0	0	1	0
writing implement	0	0	1	0	0	0
glue	0	0	0	0	0	0
scissors	0	0	0	0	0	0

Table 2: ConceptNet Matrix

Results: The results in Table 2 indicate with a "1" the presence of a tool and action relationship. While the interface of ConceptNet makes finding action-tool relationships simple, we did not use the ConceptNet results in the final project because the data was too sparse.

Web Co-occurrence Matrix

Rather than simply having a binary matrix to indicate whether an action-tool relationship exists, we also wanted to develop an analog measure of how closely related an object and an action are. This helps us determine if one particular object is more closely related to an action than another plausible object. For example, one would think of "crayons" when "coloring", but one can color with colored pencils or markers as well. These similarity values could be extracted from any sufficiently large and relevant corpus using measures such as point-wise mutual information. Data sparsity could be an issue when working with specific domains such as Arts and crafts. However, since we only needed occurrence and co-occurrence data, we used search engine results from the web to retrieve co-occurrence data and thereby avoided data sparsity problems.

To create a matrix with these values, we used a semantic distance measure called the Normalized Google Distance (NGD). The NGD is a relatedness measure based on the Normalized Compression Distance (NCD) and modified to use the number of results returned by Google to stand in for the compressor in NCD (Cilibrasi and Vitanyi, 2007). In this project, we use Yahoo instead of Google because the Yahoo API is more flexible. To find the semantic distance between two terms x and y, we use the Normalized Google Distance equation:

$$NGD(x,y) = \frac{\max\{\log f(x), \log[f(y)\} - \log f(x,y)]}{\log(N) - \min\{\log f(x), \log[f(y)\}]}$$

Equation 1: Normalized Google Distance

where $f(x)$ is the number of search results for query x, $f(x,y)$ is the number of search results for query "x and y ", and N is the total number of pages indexed by the search engine. This theoretically returns a value between 0 and infinity (because of how Google calculates the number of results), but most values returned are between 0 and 1.

The lower the number, the more related two queries are, with two identical queries having a distance of 0. We made sure that the verbal the -ing form of the verb was used in order to discriminate the verb from the noun form. We calculate the NGD for each action-tool pair and enter it into a matrix to form our co-occurrence matrix.

Domain-Specific Comparison: Action-tool relationships differ according to a particular domain. For example, in Arts and crafts, one associates “cutting” with “scissors”. However, in cooking, one would associate “cutting” with “knife”. We needed to capture these domain-specific relationships to improve the accuracy of our reported similarity. To restrict to a particular domain using the NGD, we appended the domain to each of our queries. For example, to determine *f*('scissors'), we performed the query <scissors “arts and crafts”>. Likewise, to determine *f*('scissors’,‘cutting'), we performed the query <scissors cutting “arts and crafts”>.

Adding the domain to the search query did not completely disambiguate domain-specific relationships. Even if we specify the domain as “arts and crafts”, we obtain a smaller distance between “knife” and “cutting” than we do between “scissors” and “cutting”. We further enforced domain specificity by multiplying our distance by a scaling term, which is the distance between the object and the domain. Thus, if a particular object is not associated with the domain, it will be given a larger distance. The equation is then as follows:

$$SNGD(x, y) = NGD(x, y) * NGD(x, domain)$$

Equation 2: Domain Scaled Normalized Google Distance

where x is the object and y is the action.

Pattern Matching: Our implementation so far has been searching for a co-occurrence of two words anywhere in an entire web page. We needed to discard the incidental matches in our search query, and work under the assumption that two words are related only if they occur within a certain word distance within each other. We achieved this by using the pattern matching feature provided by the Yahoo API. This pattern matching feature allowed us to perform similar queries to those we could make with AltaVista's “NEAR” operator. If we change our query to < “scissors * cutting” OR “cutting * scissors” “arts and crafts” >, the search only returned pages where “scissors” and “cutting” appear within two words of each other. This was preferable to searching for words that are adjacent to each other, given that there are often prepositions or other words in between a tool and an action. We found that a distance of two words provided results that most coincided with our expectation of co-occurrence.

	color	cut	draw	glue	paint	place
brush	2.51	2.11	2.40	6.17	1.85	6.17
writing implement	2.12	3.51	1.72	6.17	2.08	6.17
glue	2.51	2.51	2.51	1.20	2.44	6.17
scissors	2.47	1.76	2.36	6.17	2.68	6.17

Table 3: Web Co-Occurrence Matrix

Results: Our results for the Web matrix correlate with both our expectations and the co-occurrences of the training data. Table 3 shows the final matrix that was used in the global model. In this matrix, the action-tool pairs that we expect to see in video have the lowest distance values. In this case, values below 2.0 indicate that an action-tool relationship exists, and this pattern provides the co-occurrence information that can be used in the global model.

Related Research

Ikizler-Cinbis et al. (2010) is somewhat related in that it combines images collected from the web in an iterative process in order to obtain cleaned images that will be used to annotate videos. In our work, instead of using images, we collected concepts and relationships to annotate videos. Kojima et al. (2010) address the semantic gap between video and texts; they propose a method for generating textual descriptions of human behaviors appearing in video using semantic features of human motion which are associated with the concept hierarchy of actions.

Relatedness measures have often been used to find relationships between words. While a relatedness measure can be used to find synonymous words, relatedness (as opposed to similarity) is also used to find words that share any association, such as automobile and gasoline. Resnik (1995) and others have used WordNet in conjunction with information content from other corpora to determine word similarity. However, WordNet's relationships are limited to the same part of speech and we found that WordNet similarity measures could not be used to find action-tool relationships.

Wikipedia, given the vast amount of information it contains, has been used for relatedness measures. Using machine learning techniques, Gabrilovich and Markovitch (2007) generated semantically related Wikipedia concepts from a given input word. This system utilizes the same assumption we do for semantic relatedness – that a large majority of words in a Wikipedia article will be related to the subject of the article.

Web-based relatedness measures have also been explored in previous work. Baroni and Vegnaduzzo (2004) used a variant of point-wise mutual information with AltaVista using the “NEAR” operator, which only returns searches that contain the specified words within a certain word distance of each other. Cilibrasi and Vitanyi (2007) developed the Normalized Google Distance (NGD) to calculate relatedness based on Google search results. In our work, we used a modified NGD to weigh relatedness based on domains (such as "arts and crafts") and word proximity. These web-based relatedness measures can also be modified to utilize existing knowledge bases. Gracia and Mena (2008) modify the NGD to use in word sense disambiguation and to calculate a relatedness score given two words from different ontologies. In our project, we are given a domain based on the content of videos rather than an ontology, and this serves to disambiguate possible conflicting senses.

Integration with the Vision Models

In our application we demonstrate how to combine contextual priors automatically extracted from transcripts and web mining with multiple visual cues such as motion descriptors, object presence, and hand poses. Using these diverse features and linguistic information we make use of several increasingly complex computational models for recognizing elementary manipulation actions and composite activities. The final video annotation problem is formulated in terms of a chain conditional random field (CRF) where individual states correspond to shots of elementary manipulation actions. The detailed aspects of the visual features and the CRF models are described in another publication, thus we will restrict ourselves to a discussion of the results in this paper (Anonymous, 2011).

Joint Modeling of Tools and Actions

Due to the complex nature of the data, we proposed to use several low and mid-level features extracted from videos to aid the classification. These features included local motion signatures, the absence and presence of specific object categories, hand poses and domain specific contextual priors. These features were integrated in a single shot action recognition model. This joint model captures explicit dependencies between the features in a CRF framework. Table 4 shows results from our structured joint modeling CRF experiments.

Normalized Accuracy	No joint modeling (visual only)	Ground-truth Co-occurences	Domain Knowledge Modeling from Text
Action	50.9	50.8	50.8
Tool	44.9	40.7	48.3
Action + Tool	28.0	40.7	37.8

Table 4: Accuracy of joint feature model

In column 1, we do not use any information extracted from text, and we can see that although we recognize either the Action or the Tool, we do not do well when recognizing the pair of them that do co-occur. We found that explicitly modeling the co-occurrence of actions and tools either directly using our ground truth (column 2) or using text-based domain knowledge (column 3) significantly helped results. For domain knowledge, we learned a weighted combination of the action-tool co-occurrence matrices that we described before. The action accuracy (row 1) remained nearly constant throughout experiments, but the tool accuracy (row 2) and accuracy in getting the correct tool and action together in the same example (row 3) increased significantly when modeling action-tool co-occurrence.

Most importantly, these results demonstrate that it is possible to "plug in" domain knowledge as a substitute for ground truth information and get comparable performance gains over using no joint modeling.

Temporal Modeling

The previous section we described how the co-occurrence information that we extract from the web, can be used to improve the joint estimation of actions and tools. Since we are not only interested in annotating individual segments of a show independently, we also explored how to make use of the temporal information contained in the text to increase our annotation accuracy.

In our case, we hypothesized that the relative order of action verbs in the transcript is highly correlated with the relative order of actions in the video, i.e., if one action is mentioned before another action in the text, the probability of finding the corresponding actions appearing in the same order in the video should be higher than the opposite case when their order is reversed. Since the text and video are not strictly aligned in real videos, we do not restrict ourselves to only ordering constraints between direct neighbors, but look at all pairs up to two positions apart.

In our experiments, we were interested in the relative performance change in action classification accuracy, i.e., with and without prior temporal knowledge obtained from transcripts and online instructions. In Table 5 we summarize the performance of average classification accuracy for the PBS Sprout TV dataset.

Motion features only (state of the art base line), no temporal info	0.42
Joint Modeling of Motion, Tool, and Hand Features, no temporal info	0.47
Joint CRF Model, no temporal info	0.51
Temporal CRF with transcript input	0.52
Temporal CRF with instruction input	0.53

Table 5: Overall action classification accuracy for the Sprout TV dataset.

These overall results show that incorporating additional features as well as endowing the model with more explicit structure and priors proved beneficial and yielded improvement in overall accuracy from 42% to 53%. Furthermore, the temporal knowledge extracted from transcripts and online instructions further improved the classification performance compared to the models without temporal information. Comparing the two types of text sources, online instructions are slightly more helpful than transcripts. This might be because transcripts are noisier than online instructions since they contain large amount of narration. Consequently, the verb list extracted from transcripts contains more irrelevant verbs besides the action verbs we are interested in.

Conclusion and Future Work

The contribution of this work has many aspects. We explored the optimal representation for language models in (a) parsing, extracting, and filtering textual information, (b) exploiting publicly available knowledge sources and the web, and (c) integrating these representations with the overall vision system. We demonstrated that modeling the co-occurrence of actions and tools using text-based domain knowledge significantly helped results. The approaches presented here are fully independent of the domain, and the system is completely scalable to other types of actions. In future work, we will explore further the semantic action primitives and will associate them to clusters of frames corresponding to minimal actions.

References

Baroni, Marco, and Stefano Vegnaduzzo. Identifying subjective adjectives through web-based mutual information. In Proceedings of KONVENS, 7th German Conference on Natural Language Processing, 2004.

Cilibrasi, Rudi L. and Paul M.B Vitanyi. The Google Similarity Distance. IEEE Trans. Knowledge and Data Engineering, 19(3), 2007.

Fellbaum, Christiane. WordNet: An Electronic Lexical Database.MIT Press, 1998.

Finkel, Jenny Rose, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by Gibbs sampling. In Proceedings of the 43rd Meeting of the Association for Computational Linguistics, 2005.

Gabrilovich, Evgeniy and Shaul Markovitch. Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In Proceedings of The Twentieth International Joint Conference for Artificial Intelligence, Hyderabad, India, 2007.

Gracia, Jorge and Eduardo Mena. Web-based measure of semantic relatedness. In Proceedings of 9th International Conference on Web Information Systems Engineering, volume 5175, 2008.

Ikizler-Cinbis, Nazli, R. Gokberk Cinbis, Stan Sclaroff. Learning Actions From theWeb. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, 2010.

Klein, Dan and Christopher D. Manning. Accurate unlexicalized parsing. In Proceedings of the 41st Meeting of the Association for Computational Linguistics, 2003.

Kojima, Atsuhiro and Takeshi Tamura Kunio Fukunaga. Natural Language Description of Human Activites from Video Images Based on Concept Hierarchy of Actions. International Journal of Computer Vision, Kluwer, 2001.

Liu, H. and P. Singh. Conceptnet - a practical commonsense reasoning tool-kit. BT Technology Journal, 22(4):211-226, Oct 2004.

Resnik, Philip. Using information content to evaluate semantic similarity in a taxonomy. In Proceedings of IJCAI, Montreal, Canada, 1995.

Ruiz-Casado, Maria, Enrique Alfonseca and Pablo Castells. Automatic extraction of semantic relationships for WordNet by means of pattern learning fromWikipedia. Natural Language Processing and Information Systems, Lecture Notes in Computer Science, 2005.