Technical Disclosure Commons

Defensive Publications Series

April 2020

Multi-Modal Visual and Memory Coreference Resolution

Anonymous

Multi-Modal Visual and Memory Coreference Resolution

ABSTRACT

An automated assistant in an augmented reality (AR) device or smartphone performs coreference resolution. The automated assistant resolves references that are mentioned in a user's dialog, e.g., audio input, by analyzing the audio input, visual input, and stored information that represents the user's memory. The automated assistant performs the coreference resolution so as conduct an intelligent dialog with the user for shopping, visual question answering (VQA), or other interactive activity.

KEYWORDS

- Coreference resolution
- Augmented reality (AR)
- Visual question answering (VQA)
- Visual dialog
- Automated assistant
- Artificial intelligence (AI)

BACKGROUND

In computational linguistics, coreference resolution refers to the process of resolving the references, e.g., nouns and pronouns, that are mentioned within a textual or audio dialog. A goal of coreference resolution is to properly interpret a phrase and context of the phrase, by determining which pronouns or nouns in the phrase are referring to each other or to particular subjects.

An example phrase (which is used throughout this disclosure) might be the following, which is spoken by a user while viewing a product during online shopping: “Is this brown couch going to fit in our living room along the window?”

In the foregoing example, existing textual/audio coreference resolution techniques may generally be able to determine that “this” refers to “brown couch” (for “this brown couch”) and that “our” refers to “living room” (for “our living room”). However, existing coreference resolution techniques are sometimes inaccurate and generally incapable of understanding the scene being viewed by the user, mapping the user’s language (audio) to the “brown couch” being viewed, determining which living room or window that the user is thinking of, etc. Therefore, existing coreference resolution techniques are insufficient to enable the user’s question of “Is this brown couch going to fit in our living room along the window?” to be answered.

DESCRIPTION

This disclosure describes a coreference resolution module that is implemented in an augmented reality (AR) device (e.g., AR glasses) or in some other user device (e.g., a smartphone). The coreference resolution module is multi-modal in that audio input, image input, and stored information (representing the user’s memory) are used to perform coreference resolution. Data collection processes are performed to build annotated data sets that represent the user’s memory and that are used to train the coreference resolution module.

In operation, the coreference resolution module is implemented by an automated assistant or other form of artificial intelligence (AI) module in the user device. Thus, when the user of the user device is viewing a scene (e.g., an AR scene through the AR glasses, or a website or other image displayed on a screen of the smartphone), the user may provide an audio input (e.g., “Is

this brown couch going to fit in our living room along the window) to the automated assistant. The coreference resolution module of the automated assistant attempts (a) to resolve the references in the audio input (e.g., associate “this” with “brown couch”, “our” with “living room”, etc.), (b) to map these references to the scene being viewed by the user (e.g., identify the “brown couch” in the scene provide by video input), and (c) to identify one or more relevant stored datasets (e.g., previously stored photos of the living room and/or the window, which represent the memory of the user) that are indicative of what the user is thinking of.

Given the above three inputs of video, audio, and stored data, the coreference resolution module generates multiple mention-referent (m, r) pairs, and ranks each of the (m, r) pairs, with the highest-ranked (m, r) pairs representing the resolved coreferences. The automated assistant uses the resolved coreferences to intelligently perform visual question answering (VQA) or a visual dialog with the user, e.g., to engage in a conversation with the user to determine whether the couch will fit in the living room along the window.

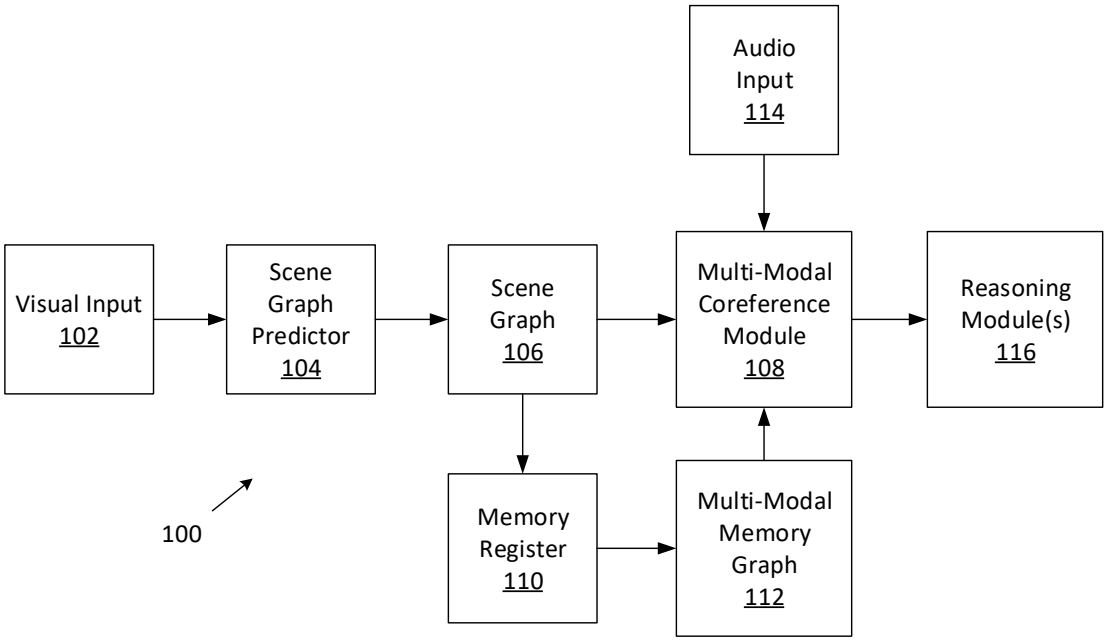

FIG. 1: Performing multi-modal coreference resolution

FIG. 1 is a functional block diagram a system (100) that performs a method for multi-modal coreference resolution. The system receives visual input (102), in the form of a video or other image. For instance, the image may depict a scene that is being viewed by a user of AR glasses, an image posted on a web page being viewed by the user, or other type of visual input. In the current example, the image is that of a couch on a web page of a furniture vendor, an actual couch that the user is viewing with AR glasses at a furniture store, or a virtual image of a couch in some room rendered by AR glasses, etc.

The visual input is provided to a scene graph predictor (104), which extracts the objects in the image and predicts their relationships. For example, the scene graph predictor analyzes the image to identify the couch, its color (e.g., brown), its orientation and properties (e.g., running lengthwise, with sides and a back, leather texture, etc.), its spatial relationship with other objects in the image (e.g., the couch is standing on a floor), etc.

The scene graph predictor generates a scene graph (106), which is a data structure that arranges the logical and spatial relationships of the objects in the image. For example, information such as couch, brown, leather texture, floor, etc. form the branches, nodes, and leaves of a tree of the scene graph. The scene graph is provided as one input to a multi-modal coreference module (108).

Furthermore, the scene graph is stored and indexed in a memory register (110) along with other datasets that represent the memory of the user, with permission from the user. In this example, the user has viewed an image of a brown couch, and so the brown couch has become one of the recollections/ remembrances in the memory of the user. With user permission, the memory register also stores other datasets of experiences and actions of the user: calendar events, email content, photos, contact lists, recent phone calls/voicemails, web browsing history

(e.g., recently viewed items), other scene graphs previously generated by the scene graph predictor, etc., all of which form the body of knowledge that make up the memory of the user. Thus, in the current example, the memory register stores and indexes a scene graph or a photo that shows the "living room" and the "window." The content(s) of the memory register can be used to generate a multi-modal memory graph (112), which serves as another input to the coreference module, in a manner that is described later below.

Another input to the coreference module is audio input (114). In the current example, the audio input is the question "Is this brown couch going to fit in our living room along the window?" that the user asks the automated assistant while viewing the image of the brown couch.

In operation, the coreference module receives the audio input and the scene graph for the visual input. The coreference module parses the audio input, and generates and ranks all possible (m, r) pairs from the words in the audio input, with "this" and "couch", "our" and "room", "brown" and "couch", "living" and "room", etc. being the (m, r) pairs that are ranked higher relative to other possible (m, r) pairs, due to their accuracy/correctness in this example.

To map the visual input and the audio input with each other, the coreference module traverses the branches/nodes/leaves of the scene graph to find the elements that pertain to the brown couch. When these elements are located in the scene graph, the coreference module is able to conclude (e.g., resolve) that the "brown couch" uttered in the audio input co-refers to the brown couch that appears in the visual input.

Since the elements of the living room and the window do not appear in the visual input, the coreference module does not find any related branches/nodes/leaves of the scene graph that

was generated from the visual input. Therefore, the coreference module attempts to resolve these elements from one or more of the memory graphs generated from the datasets stored in the memory register. For example, the coreference module can query the memory register for datasets (e.g., stored scene graphs, photos, emails, etc.) where the terms "living room" and "window" are indexed. The coreference module receives the results of the query, and then generates (m, r) pairs or other types of relationships that link the "living room" and "window" from the memory register, with the contents of the audio input and/or the visual input.

Having thus completed the coreference resolution to link the contents of the visual input, the audio input, and the memory register with each other, the coreference module provides the linked contents to one or more reasoning module(s) (116). The reasoning module(s) uses various tools, algorithms, and data to answer the user's question "Is this brown couch going to fit in our living room along the window?"

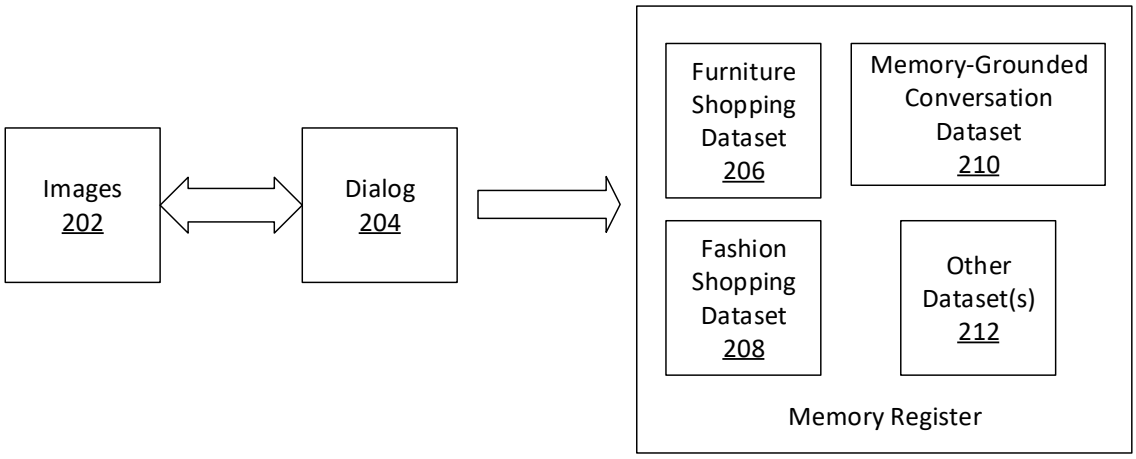

FIG. 2: Training the coreference module and building datasets

FIG. 2 shows a training process to train the coreference module and to build the datasets that are stored in the memory register of FIG. 1. The automated assistant engages in visual question answering (VQA) or visual dialog with the user, in which the automated assistant

presents images (202) to the user, and the user and automated assistant conduct a dialog (204) pertaining to the images, represented by the double-headed arrow in FIG. 2. The dialog results (represented by the single-headed arrow in FIG. 2) in the generation of scene graphs, memory graphs, and other types of indexed and/or structured data that may be stored in the memory register for later use as reference when performing coreference resolution in the manner described previously above, as well as for continuously training the coreference module.

A first example of a dataset generation process involves the user engaging in a furniture shopping dialog with the automated assistant. The user says, "I want to see some couches." Then, the automated assistant presents an image carousel of couches that are available from a website of a vendor, and asks "How about these couches?" Next, the user replies with "I like the brown one, next to the blue one. Can you give me more details?" The automated assistant then presents the user with multiple views of the brown couch, along with information provided by the website, such as price, size, material, etc. A log is created during the course of the dialog, and the contents of the log (e.g., terms like couch, brown, leather, etc.) are used to generate a scene graph that is stored as a furniture shopping dataset (206). Other metadata associated with the dialog are also stored in the memory register, such as date/time that the website was accessed, items viewed, product details about the viewed items, etc., and are usable for reference to the user's memory.

A second example of a dataset generation process involves the user engaging in a fashion shopping dialog with the automated assistant, in which fresh images plus data from the user's memory are used during the dialog. The automated assistant presents a seed image to the user, such as that of a blue coat, and the user responds by saying "I love this blue coat! Is it more expensive than the red coat I saw earlier?" The automated assistant then checks the contents of

the memory register for references to the "red coat" (including contents of previously viewed web pages or other content), and then presents an image of the red coat to the user along with a statement like "The red coat you saw earlier is $250, whereas this blue coat is $200 and does not come in size M." The user responds with "Do you have a coat in size M?" The automated assistant then presents a carousel of coats in size M that are available, in newly presented images or in previously presented images from the memory register. As before, the dialog is logged as multiple images are presented to the user, and the logs are used to generate one or more fashion shopping datasets (208).

A third example of a dataset generation process involves the user engaging in a memory-grounded conversation with the automated assistant. In this process, natural sounding dialogs are generated between the user and automated assistant, based on stored images/photos that are presented to the user, and the automated assistant introduces further images to elicit a continued dialog with the user. The user initially asks, "When was the last time I went fishing?" The automated assistant then searches stored photos or other stored information for "fishing", and presents a photo and responds with an answer of "It was July 2019 with John. Do you want to see other fishing photos with John?" If the user responds with "Yes", then the automated assistant locates and presents the other photos and also says "Here are some photos from 2018." The conversation can continue with the user asking, "Who else joined in the fishing trips in 2018?", and the automated assistant can determine names from annotations in the photos, from email, etc. and present the names to the user, and the conversation can continue. The full conversation is logged, and stored as a memory-grounded conversation dataset (210) in the memory register.

Other dataset(s) (212) that result from dialogs between the user and the automated assistant can be stored in the memory register, for training purposes and also for reference during the course of performing coreference resolution. Other data such as browsing history, calendar events, web pages, emails, etc. may also be stored for training and/or coreference resolution purposes, with user permission.

CONCLUSION

The techniques described herein enable an automated assistant to perform coreference resolution to link elements contained in visual input, in audio input, and in stored information that represents a memory of a user. The automated assistant performs a training process to build/collect datasets that are stored as memory references of the user and that are used while performing the coreference resolution.