November 2022

Contact Sharing with Automatic Format Translation

Dhandapani Shanmugam

Contact Sharing with Automatic Format Translation

ABSTRACT

Different users have different preferences when storing contacts in an address book. For example, conventions used to name contacts include concatenations of first and last names (‘AliceMiller’); names of relationships (‘Mom’); hierarchies (‘theFirm.Owner.Joe’); etc. When a user receives contact information from another user, manual editing is required to bring it to their preferred format. This disclosure describes techniques, implemented with user permission, to infer the preferred contact naming format of a user and to automatically transform newly received contacts to the preferred format. User and contact metadata are used to build an ontology that maps relationships between the user and their contacts. A newly received contact is fit into the ontology to infer the relationship of the newly received contact to the user. Based on the inferred relationship, the user is offered an option to store the newly received contact reformatted based on their preferences.

KEYWORDS

- Naming convention
- Contact information
- Contact format
- Address book
- Contact list
- Phone contacts
- Contact sharing
- Contact exchange

BACKGROUND

User devices such as phones, tablets, etc. enable users to store the names, phone numbers, and other contact information in an address book, also known as a contact list. Contacts can be stored on-device or synced with the user's cloud account. The user can store a variety of details for a contact, such as first name, last name, photo, home phone number, mobile phone number, email address, website, birthday, company, relationship, a free form note, etc.

Many users have the habit of storing contacts in a form convenient and familiar to them. For example, users often use personal conventions for naming, e.g., concatenating first and last names in the first-name field ('AliceMiller'); using a relationship in the first-name field ('Mom' for 'Alice'); etc. Another frequent naming convention is a hierarchical name-structure, e.g., prefacing the name of a user with their organization ('theFirm Bob'), title ('theFirm Owner Bob'), etc. Similarly, users have preferred languages for storing contacts, preferred photos for contacts, etc.

The free-form nature of fields in the address book, while providing choice for the user, can lead to problems when a user shares their contacts with other users. For example, a user Bob might use the phrase 'Dad' in the firstname field of a person named Joe. Bob's mother Alice might prefer to use a different phrase 'family.husband.Joe' (or simply 'Joe') in the firstname field of the same person Joe. When Alice sends Joe's contact information to Bob, Bob must manually edit the information to bring it to a format convenient and familiar to him. The need for manual editing of contact information, which is time-consuming and error-prone, is felt almost any time a contact is exchanged between users that don't share the same contact naming conventions or relationship with the contact.

Contacts or address book applications do not contextualize a shared, extracted, or imported contact to automatically store it in a format that is convenient and familiar to the user.

DESCRIPTION

This disclosure describes techniques to infer the preferred format of stored contacts of a user and to automatically transform newly received contacts to the preferred format. The techniques are implemented only upon specific permission from the user. User data, such as contacts/address book, is accessed and utilized for the purpose of determining formats and format conversion for contacts. The techniques are implemented locally on the user device or can optionally be implemented in the cloud if the user permits.

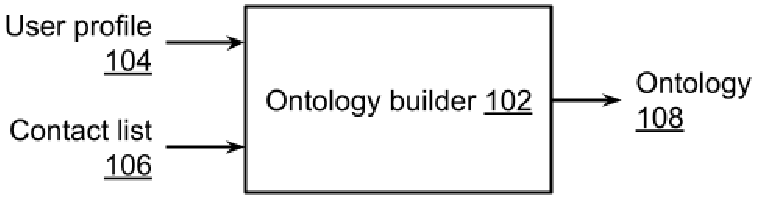

Fig. 1: Building an ontology from a contact list

As illustrated in Fig. 1, the user profile (104) and their contact list (106) are provided as input to an automatic ontology builder (102) to create an ontology (108). The ontology is a graph of relationships between the user and their contacts. The automatic ontology builder can be, e.g., a machine learning model trained to recognize relationships between objects (e.g., contacts) given metadata (e.g., phrases within fields of the contact) about the objects. With user permission, metadata of the contacts and of the user such as their public profiles on social media, video/photo sharing websites, or other sources of public profiles can be used to build the ontology. Effectively, the ontology builder transforms a flat list of contact information into rich, graphically structured data.

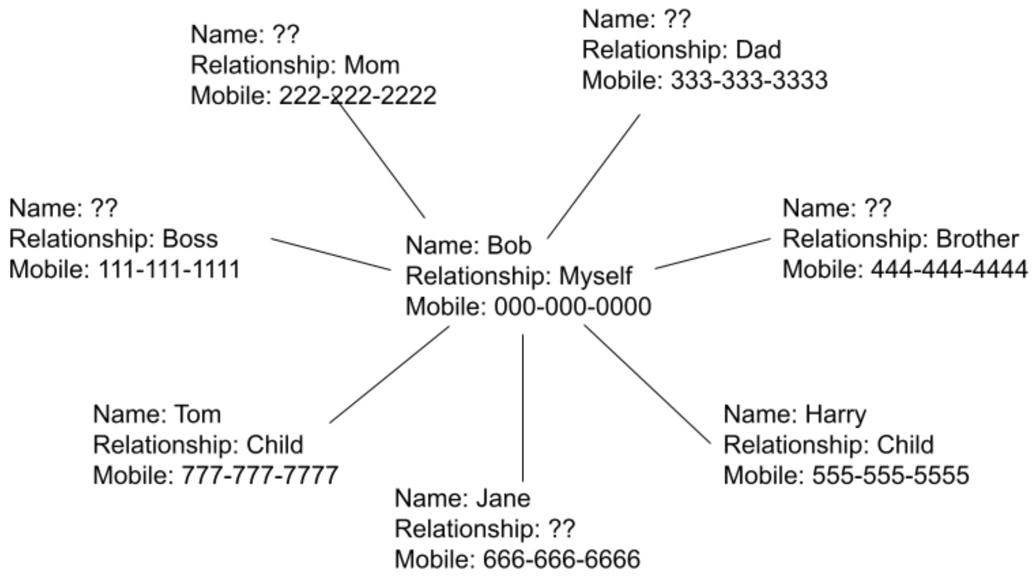

Fig. 2: An example ontology

Fig. 2 illustrates an example ontology. From the user profile, it is determined that the name of the user is ‘Bob.’ The user Bob generally prefers to store his contacts by relationship, such that some of Bob’s contacts (e.g., ‘Mom’, ‘Dad’, ‘Brother’, ‘Boss’) lack names, but their relationship to Bob is known or is inferred by the ontology builder. On the other hand, some contacts (‘Jane’) have an unknown relationship to Bob, while for still other contacts (‘Tom’, ‘Harry’) both names and relationship to Bob is known.

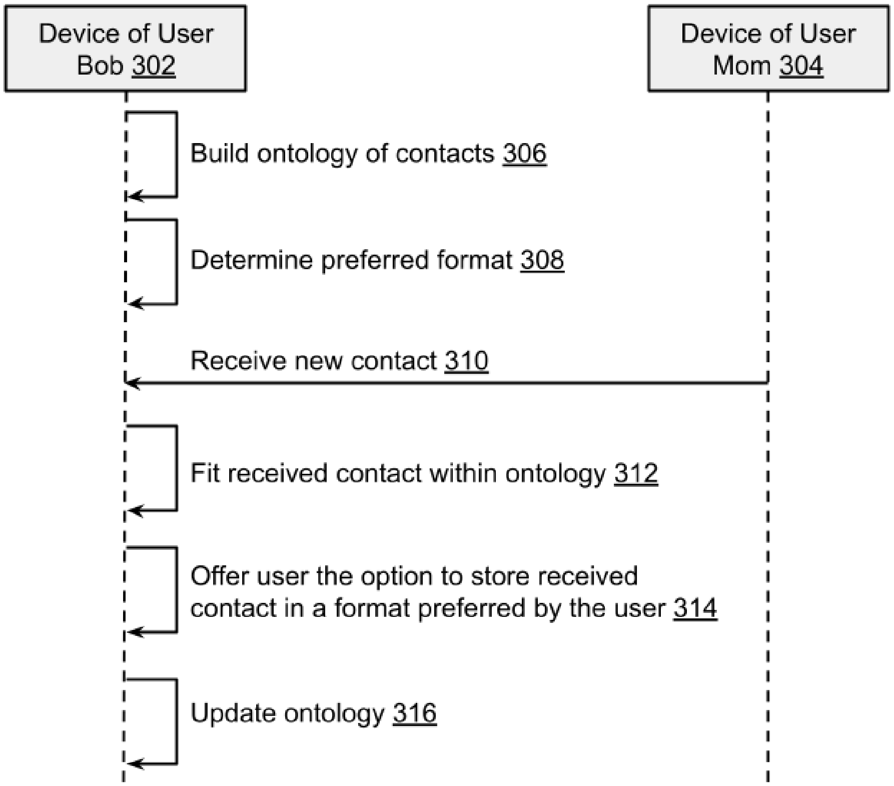
Device of User Bob 302
Device of User Mom 304
Build ontology of contacts 306
Determine preferred format 308
Receive new contact 310
Fit received contact within ontology 312
Offer user the option to store received contact in a format preferred by the user 314
Update ontology 316

Fig. 3: Reformatting a received contact to a form familiar to the receiving user

Fig. 3 illustrates reformatting a received contact to a form familiar to the receiving user. As explained earlier, an ontology of contacts is built (306) for the user Bob using a user device (302). The format for contacts preferred by the user Bob is also determined (308). For example, it can be determined that the user Bob prefers to store relationship names (e.g., mom, dad, boss) in the first-name field of his contacts. Next, the user Bob receives a contact (310) from a user stored as 'Mom' (304) in Bob's contact list. The received contact is originally formatted in a manner set by the user Mom as follows.

```
First Name: family.husband.joe
Number: 888-888-8888
```

The received contact is analyzed by the ontology builder and fit into the existing ontology (312). For example, the phrase ('family.husband.joe') in the first-name field of the received contact can be used to infer that there is a high likelihood that the received contact, with first name ‘family.husband.joe,’ is the same as the contact ‘Dad’ in Bob’s contact list, and that the contact ‘Dad’ in Bob’s contact list is named ‘Joe.’ The ontology can be updated as shown in Fig. 4 below. Recognizing that the received contact with first name ‘family.husband.joe’ is likely the same as the existing contact ‘Dad’ in Bob’s contact list, the user Bob is offered the option to store the received contact in the format generally preferred by him (314), e.g.,

```
First Name: Dad
Number: 333-333-3333
Number: 888-888-8888
Notes: Joe, father
```

If the user accepts the option that ‘family.husband.joe’ and ‘Dad’ are one and the same, the contacts are effectively merged. The updated ontology of Fig. 4 is confirmed as reliable (316).

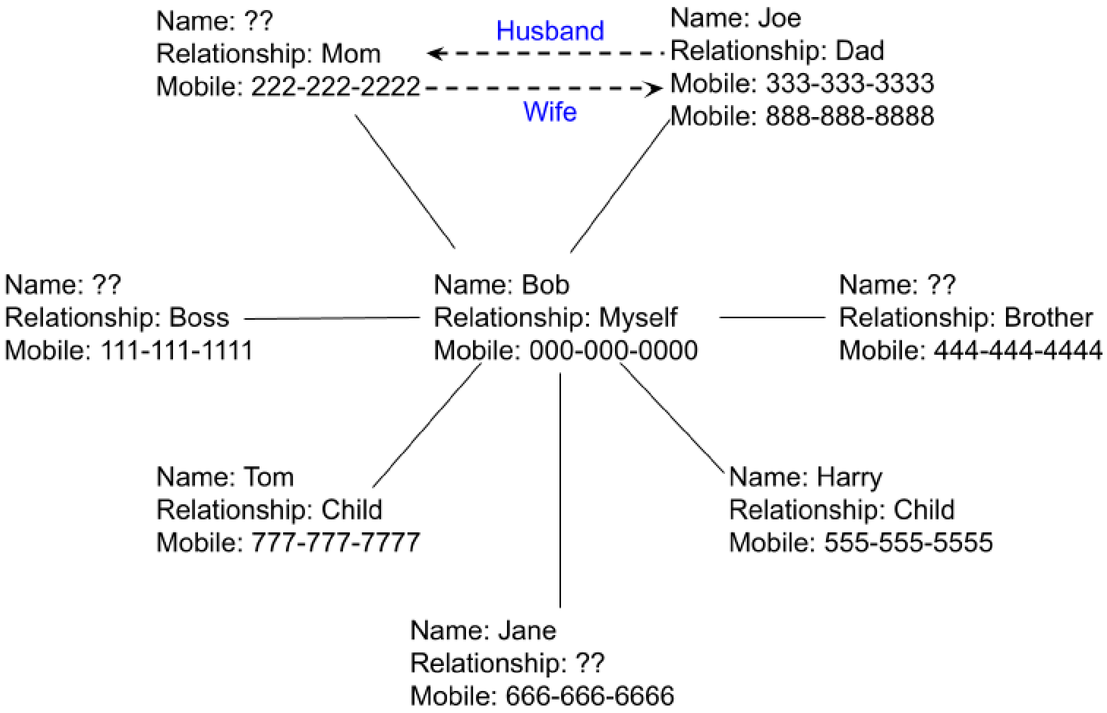

Fig. 4: Ontology update on the basis of a newly received contact

Alternatively, recognizing that the received contact with first name ‘family.husband.joe’ is in fact different from the existing contact ‘Dad’ in Bob’s contact list, the user Bob is offered the option to store the received contact as a distinct, new contact, as follows.

```
First Name: Joe
Number: 888-888-8888
Notes: Joe, husband of Mom
```

If the user accepts the option that ‘family.husband.joe’ and ‘Dad’ are distinct contacts, then the ontology is updated as illustrated in Fig. 5.

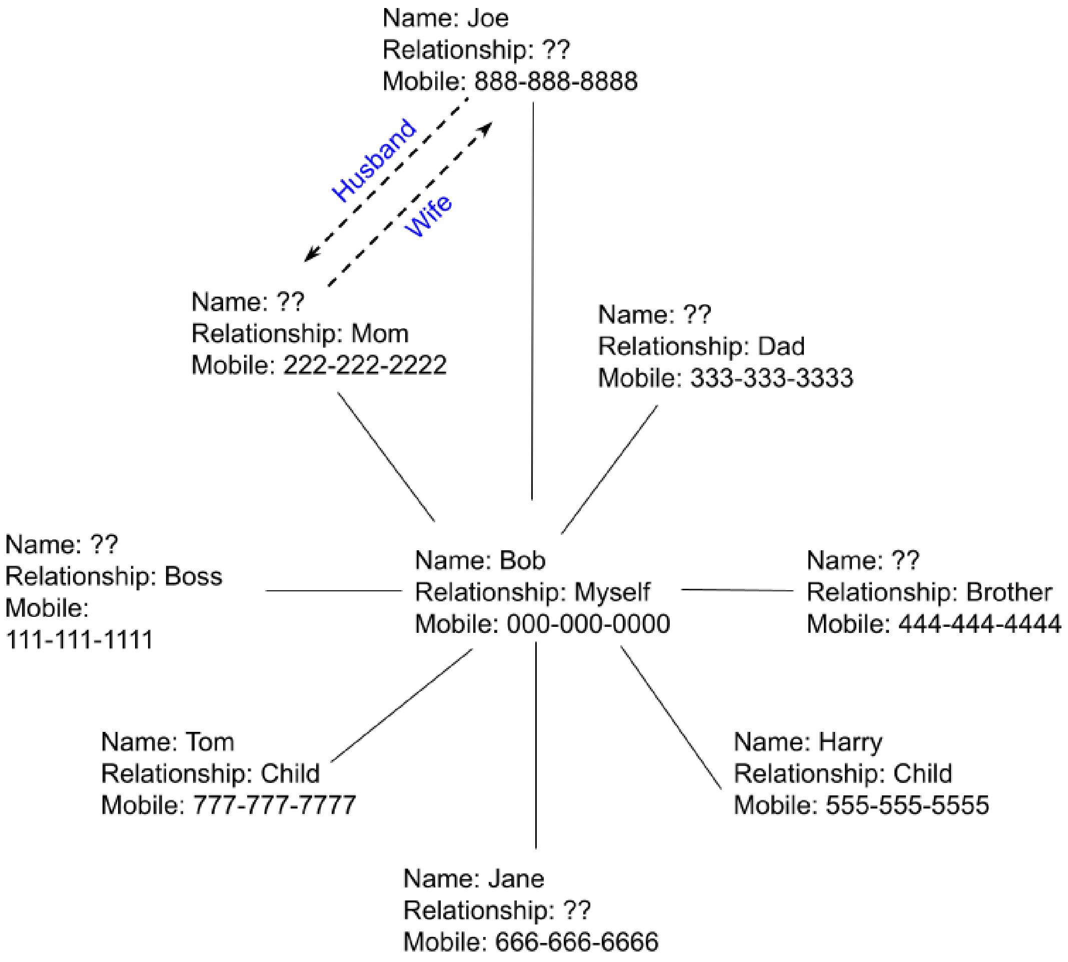

Fig. 5: Alternative updated ontology based on user selection

The received contact can be automatically stored in the preferred language of the user. Furthermore, with user permission, the public profile (e.g., in photo/video sharing apps, social media platforms, etc.) of the received contact can be accessed in the context of the relationship between the user and the received contact to identify a photograph of the received contact. With user permission, the photograph of the received contact can automatically be incorporated into the address book. For example, the contact 'Dad' being determined as identical to 'family.husband.joe,' a shared photo album can be scanned to retrieve an image of 'family.husband.joe,' which can then be used to represent the contact 'Dad' in Bob's address book.

In this manner, the described techniques simplify the translation of contact naming conventions between users and simplify the sharing of contacts between different users. The format of contacts created, stored, or maintained in an address book of a mobile device or app is determined and utilized for the automatic translation of contact information to match the user preferences of the user receiving the contact information.

Further to the descriptions above, a user may be provided with controls allowing the user to make an election as to both if and when systems, programs, or features described herein may enable the collection of user information (e.g., information about a user's social network, a user's address book/ contacts, social actions or activities, profession, a user's preferences, or a user's current location), and if the user is sent content or communications from a server. In addition, certain data may be treated in one or more ways before it is stored or used so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be determined for the user, or a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code, or state level) so that a particular location of a user cannot be determined. Thus, the user may have control over what information is collected about the user, how that information is used, and what information is provided to the user.

CONCLUSION

This disclosure describes techniques, implemented with user permission, to infer the preferred contact naming format of a user and to automatically transform newly received contacts to the preferred format. User and contact metadata are used to build an ontology that maps relationships between the user and their contacts. A newly received contact is fit into the ontology to infer the relationship of the newly received contact to the user. Based on the inferred relationship, the user is offered an option to store the newly received contact reformatted based on their preferences.

REFERENCES

[1] Michal Szczerbak, François Toutain, Ahmed Bouabdallah, and Jean-Marie Bonnin. "Collaborative context experience in a phonebook." In *WAINA 2012: 26th International Conference on Advanced Information Networking and Applications Workshops*, pp. 1275-1281. IEEE, 2012.