Knowledge and Belief by K M Tawsik Jawad

Knowledge – Belief Flow:

Agents Selected: Among the diverse types of intelligent agents available, knowledge-based agents are chosen. This is done because the architecture is designed for inter agent communication which is ideally suited for knowledge-based agents. Three agents here are: User Analyzer, User Group Analyzer and Topic Assigner agent. Out of them User Group Analyzer agent takes in the user clusters provided which is the output of agent 4 implemented by Atticus Beachy.

Inter-Agent Information Flow:
Agent is provided with some information to believe a certain proposition. An agent can infer certain decisions that are close to the belief it had. When an agent is provided some formulas and functionalities where it can use those formulas, it would be able to make certain predictions and analogy. That is when the belief of an agent is converted into its knowledge. In this project, the architecture of the agents is such that inter agent communication is done through knowledge sharing among knowledge-based agents.

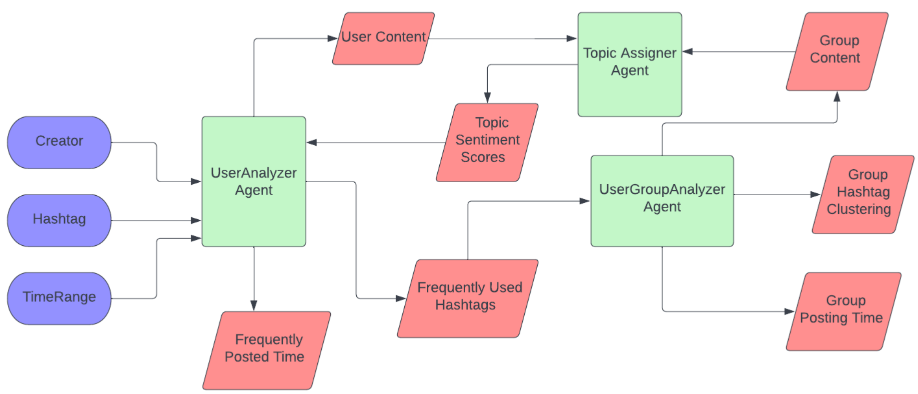

Fig 8: Inter Agent Communication Architecture for knowledge and belief

Firstly, we scrape twitter using twitter API and make a dataset with user info, tweets and tweet info. From that the most occurred users are found out so that analysis results can be interpreted definitively. The most occurred users' names are provided as input to the UserAnalyzer agent. This agent is provided with the belief that upon receiving this input, it can calculate user statistics that is frequently used hashtags by the user and frequently posted time by the user.

This belief is further extended when hashtags and timeRange are provided as inputs along with the username to the UserAnalyzer agent. So, the belief would be modified, and the the agent would filter out user tweets using 3 filters: username, hashtag and timeRange. From then on, its activity would be like what it did previously.

UserAnalyzer(Username, startDate, endDate, Hashtag) (World W1):

Belief$_{UA}$ = Given Username, can calculate user stats which include frequent hashtag retrieval and frequent time posted.

Filtering of Belief$_{UA}$ = Given Username and other parameters like dateRange or hashtags, BeliefUA will be filtered by these parameters.

Knowledge$_{UA}$ = User stats of frequent time posting, frequently used hashtags in their list of tweets.

Knowledge$_{UA(2)}$ = **(Belief**$_{UA}$**) ^ (User Statistics)**

OutputUA = Most frequent hashtags used by the user to the agent UserGroupAnalyzer. Inter agent communication from World W1 of UserAnalyzer happens in World W2 of UserGroupAnalyzer.

(Ref: Knowledge and Belief slides)

Code and Output Snippets for UserAnalyzer Agent:

```python
def main():
    df = pd.read_csv('MasterDataLoc.csv')
    userList = df['username']

    userList = userList.values.tolist()

    userList = (Counter(userList))

    mostOccuredUser = userList.most_common(1)

    for key,value in mostOccuredUser:
        username = key

    df = df[(df['username']==username)]
    htList = df['hashtags'].values.tolist()

    mostOccuredHashtag = (Counter(htList))
    mostOccuredHashtag = mostOccuredHashtag.most_common(3)

    userHashtag = "['democracy']"

    UserAnalyzer(df, username, "", "", "")

    startDate =  '2022-03-27'
    endDate = '2022-03-30'

    UserAnalyzer(df, username, startDate, endDate, "")

    UserAnalyzer(df, username, "", "", userHashtag)
```

Fig 9 : Finding out most occurred userName in the dataset

```
Tweets filtered by User libertyandfree4:


General Time Posted By User libertyandfree4
Counter({'23:38': 1, '23:11': 1, '18:32': 1, '22:43': 1, '19:32': 1, '11:00': 1,
'0:01': 1, '23:59': 1, '23:45': 1, '23:42': 1, '23:36': 1, '23:33': 1, '23:19': 1,
'23:08': 1})


Most Used Hashtags By the User: libertyandfree4

Counter({'': 11, 'democracy': 2, 'support,constitution': 1})

Tweets filtered by User libertyandfree4 and Daterange 2022-03-27 to 2022-03-30:


General Time Posted By User libertyandfree4
Counter({'22:43': 1, '19:32': 1})
```

Fig 10: User Analyzer agent output with only userName as input

```
Tweets filtered by User libertyandfree4and Hashtag: ['democracy']


General Time Posted By User libertyandfree4
Counter({'23:38': 1, '18:32': 1})


Most Used Hashtags By the User: libertyandfree4

Counter({'democracy': 2})
```

Fig 11: User Analyzer agent output with userName and hashtag given as input

Analysis of Output Snippets:

So, the most appeared user in the scraped dataset here is the username “libertyandfree4” whose statistics have been found out by the UserAnalyzer agent. It is observed that this user generally posts in the time range 23:00 to 23:59. So, this time interval might be termed as his peak tweet time.

If we analyze the hashtags used by this user, it is clear that he rarely uses a hashtag in his posts. But on a couple of occasions, he used the hashtag “democracy” also.

Extending the filtered belief and implementation of user statistics, the UserAnalyzer agent also finds out the time range in which “libertyandfree4” tweets within a particular date range. So it is seen that when the user uses the hashtag “democracy” he posts at 23:38 and 18:32 within a provided time range.

Possible 3 Worlds Inter Agent Communication:

According to the accessibility relations of possible worlds knowledge and belief, one particular agents knowledge in world W1 would be true in World W2 if world W2 is accessible to him. The vice versa relationship would occur with agent in World W2 in world W1. If it is group knowledge, then agent at a particular world having accessibility to two other worlds can gain information from both the both worlds and use that knowledge in his belief system.

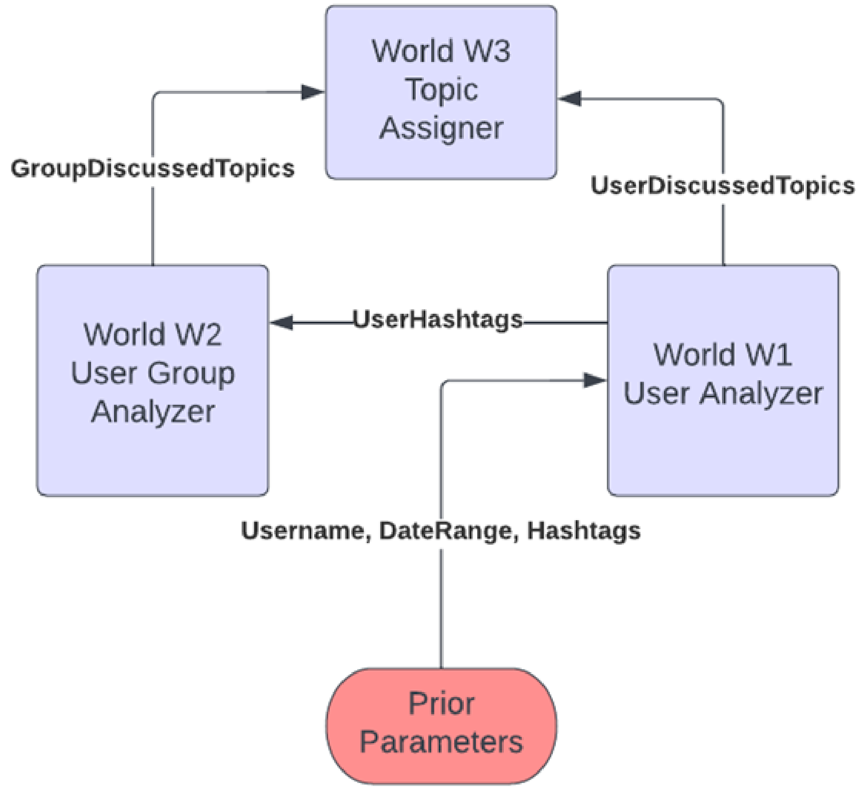

Fig 12: Accessibilty relations and inter-agent communication in the 3-world network

The diagram above represents knowledge sharing through inter-agent communication network. Initially, no common information was shared among the agents. But as the belief systems turned into knowledge and the states changed for the agents, consequently their actions determined the information flow from one agent to another. In this case, knowledge from UserAnalyzer agent is passed on in the form of userHashtags to UserGroupAnalyzer agent. But, TopicAssigner agent in world W3 has group knowledge that comes from both UserAnalyzer agent and UserGroupAnalyzer agent in worlds W1 and W2 respectively.

So, for TopicAnalyzer agent,

Input(W3): C(g, W3, W1) + C(g, W3, W2) which is the group knowledge obtained from both the worlds.

UserGroupAnalyzer (Hashtags):

Fidelity:

The userGroupAnalyzer agent is a knowledge-based agent that implements the concept of "fidelity" which is discussed in the intelligent agents slides. This agent would perform some similar actions as

UserAnalyzer because of its prior history of functions from where its input came. As a knowledge level agent following the **<See, Do, Database, Action>** tuple, it recognizes the actions performed by the agent providing its input. It takes in the process of knowledge retrieval by its input agent, saves it in the memory. When asked to perform similar actions, its history will allow it to easily perform those specified actions. The fidelity process of knowledge gain would be faster than others because information can be saved in agent's cache memory for faster retrieval.

Input: Frequently used hashtags which were the output from the agent User Analyzer.

Belief$_{UGA}$ = Use the frequently used hashtag set and haversine distance formula to calculate distance from the current user and the list of users. If the distance is within a 50km (about 31.07 mi) radius and they use the same hashtags, then those users are selected as groups of users on which fidelity actions can be performed.

Knowledge$_{UGA}$ = (Belief$_{UGA}$) * (Actions supported by fidelity of agent)

(Ref: Intelligent Agents slides)

Analysis from UserGroupAnalyzer Outputs:

```
Group Frequent Time Posting:
Counter({'5:49': 6, '22:43': 4, '14:08': 4, '17:56': 4, '18:23': 4, '17:21': 4,
'21:17': 4, '17:55': 4, '20:15': 4, '14:20': 4, '12:01': 3, '19:08': 3, '18:30': 3,
'14:53': 3, '23:56': 3, '12:25': 3, '17:45': 3, '11:03': 3, '6:52': 3, '18:08': 3,
'20:27': 3, '15:13': 3, '17:53': 3, '11:25': 3, '16:55': 3, '17:46': 3, '13:41': 3,
'15:53': 3, '23:11': 3, '18:50': 3, '13:42': 3, '15:46': 3, '11:45': 3, '11:36': 3,
'16:27': 3})
Most used hashtags by group:  Counter({'thebatman': 5, 'mlb': 5, 'amc': 4, 'batman':
4})
```

Fig 13: User Group Analyzer agents' output as group statistics

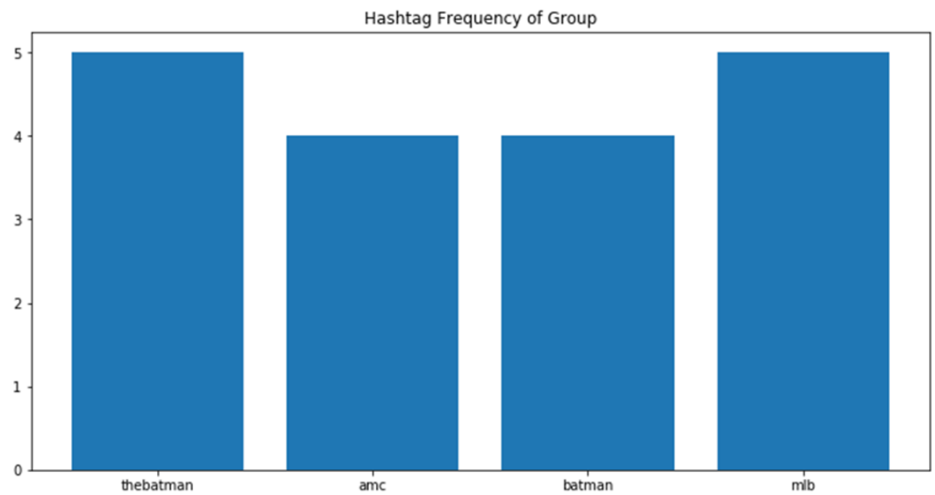

Fig 14: Frequency distribution chart of most frequent hashtags by user group

So, UserGroup is selected by the clustering technique used by Atticus Beachy in his implementation of agent 4. That is: Finding out the latitude and longitude of the user “libertyandfree4” and the frequently used hashtags of that user. Then Haversine distance is calculated between this user and all other users. If all other users' distance falls within a 50 km (about 31.07 mi) radius and they use the same hashtags as the “libertyandfree4” user, then those users are included in one group.

After finding out the userGroup, the userAnalyzer agent finds out the frequently tweeted times by the group and the frequently used hashtags by the group also. This is where the userGroupAnalyzer agent implements the fidelity concepts as discussed earlier.

Upon analyzing the time posted by the group, it’s seen that they tend to post more in early morning around 5:49 and some of them post more in the afternoon to evening interval. There are scattered time intervals as the analysis is for a group of users.

When the hashtags used by the groups are analyzed, it is evident that they use the hashtag “batman” a lot of times. This is due to the latest release of the Batman movie and the period is within this month, so users in a group commonly talk about this movie.

Topic Assigner Agent:

Belief$_{TA1}$ = Given the tweet corpus, make a vocabulary of words used in the corpus without stop words like punctuation and verbs. Then use the vocabulary to make a wordcloud of positive and negative words. Finally use 1-word or 2-word anagrams to calculate polarity of words occurring together in tweets.

Extension (Belief$_{TA1}$ **) =** Count the total number of positive, negative and neutral statements.

Knowledge$_{TA(1)}$ **= (Belief**$_{TA1)}$ **^ (Topic Probabilities in texts)**

Knowledge$_{TA(2)}$ = (Belief$_{TA1}$) ^ (Sentiment Polarity Counter)

Knowledge-Level Agents in W3: So, World W3 has two types of knowledge dedicated for Topic Probability calculation and polarity analysis of tweet corpus. Both are knowledge-level agents' abstraction concepts where abstraction is applied on their knowledge of only receiving tweet corpus, no excess detail.

Output Snippets for Topic Assigner Agent:

```
Tweets filtered by User libertyandfree4:


Topic Probabilities In Texts:  [[0.06186931 0.05765463 0.88047606]
 [0.05572935 0.8881704  0.05610025]
 [0.87898605 0.06048344 0.06053051]
 [0.88534124 0.05730974 0.05734902]
 [0.88320238 0.05853012 0.0582675 ]
 [0.08017925 0.0794206  0.84040016]
 [0.05950159 0.88183392 0.05866449]
 [0.0546817  0.89120876 0.05410954]
 [0.8895424  0.05522233 0.05523527]
 [0.88909323 0.05544892 0.05545785]
 [0.87082016 0.06441164 0.0647682 ]
 [0.06678625 0.06730735 0.8659064 ]
 [0.05812281 0.05903632 0.88284088]
 [0.05636859 0.8876198  0.05601161]]
```

Fig 15: Topic Probabilties as 1st knowledge of Topic Assigner agent

```
Sentiment_Scores :

{'neg': 0.256, 'neu': 0.684, 'pos': 0.06, 'compound': -0.9094}
{'neg': 0.153, 'neu': 0.806, 'pos': 0.04, 'compound': -0.7089}
{'neg': 0.268, 'neu': 0.607, 'pos': 0.125, 'compound': -0.8415}
{'neg': 0.09, 'neu': 0.81, 'pos': 0.101, 'compound': -0.128}
{'neg': 0.168, 'neu': 0.705, 'pos': 0.127, 'compound': -0.25}
{'neg': 0.22, 'neu': 0.636, 'pos': 0.145, 'compound': -0.3182}
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}
{'neg': 0.346, 'neu': 0.611, 'pos': 0.043, 'compound': -0.9576}
{'neg': 0.301, 'neu': 0.699, 'pos': 0.0, 'compound': -0.9514}
{'neg': 0.259, 'neu': 0.741, 'pos': 0.0, 'compound': -0.9356}
{'neg': 0.292, 'neu': 0.676, 'pos': 0.033, 'compound': -0.9075}
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}
{'neg': 0.047, 'neu': 0.891, 'pos': 0.063, 'compound': 0.2023}
{'neg': 0.083, 'neu': 0.886, 'pos': 0.031, 'compound': -0.3612}
Counter({'neg': 10, 'pos': 4})
```

Fig 16: Polarity Analysis knowledge of Topic Assigner agent

```
------- Topic Modelling for Users In a Group -------
Word-Topic Scores: ['00 innings: t1= 0.8714149937921132, t2=0.5055437739683034'
 '00 mvccbaseball: t1= 0.6921262931859535, t2=0.5044032631853561'
 '00 ve: t1= 0.7134429760805482, t2=0.5047749141545382' ...
                          'https: t1= 0.6703767348672761, t2=0.5037009210876194 שנים'
                         't1= 0.6703767348672761, t2=0.5037009210876194 :שתמיד הייתה'
                           ['uae: t1= 0.504242890450012, t2=0.6895149052954884 شكرا'

Topic Probabilities In Texts for the user group:  [[0.86535871 0.13464129]
 [0.88713587 0.11286413]
 [0.49767808 0.50232192]
 ...
 [0.09890697 0.90109303]
 [0.8757545  0.1242455 ]
 [0.11355563 0.88644437]]

Polarity Analysis of Group:
Counter({'pos': 508, 'neg': 194})

In [125]:
```

Fig 17: Word importance in topic, topic probabilty and polarity counter from group tweets

```
Tweets filtered by User libertyandfree4 and Daterange 2022-03-27 to 2022-03-30:


Topic Probabilities In Texts:  [[0.87729235 0.06088747 0.06182018]
 [0.06197921 0.87511272 0.06290807]]


Sentiment_Scores :

{'neg': 0.09, 'neu': 0.81, 'pos': 0.101, 'compound': -0.128}
{'neg': 0.168, 'neu': 0.705, 'pos': 0.127, 'compound': -0.25}
Counter({'pos': 1, 'neg': 1})
```

Fig 18: Topic Modelling and Sentiment Polarity in tweets filtered in a date range

```
Tweets filtered by User libertyandfree4and Hashtag: ['democracy']


Topic Probabilities In Texts:  [[0.05703305 0.88502614 0.0579408 ]
 [0.87746994 0.06081981 0.06171025]]


Sentiment_Scores :

{'neg': 0.256, 'neu': 0.684, 'pos': 0.06, 'compound': -0.9094}
{'neg': 0.268, 'neu': 0.607, 'pos': 0.125, 'compound': -0.8415}
Counter({'neg': 2})
```

Fig 19: Topic Modelling and Sentiment Polarity in tweets filtered with a hashtag

The 1st knowledge of topic assigner agent is obtained by first making a vocabulary of words in the tweet set of an user. Then stop words like verbs, prepositions, etc are removed from the vocabulary which does not signify any importance to the topic modelling. Then the tweet set was converted to a matrix of word count in each tweet text. This in turn was divided by the frequency of each word in the whole tweet set to calculate term frequency and inverse document frequency. The logarithmic value of this division was used to obtain the degree of belonging of each word in the topics. For simplicity, the topic range was fixed within 3 topics.

Finally, Linear Dirichlet Allocation (LDA) model was prepared to receive the word importance scores (as shown in fig 10) to calculate the degree of belonging of each tweet to a particular topic which is shown here in the figure 8.

The 2nd knowledge of topic assigner agent is obtained by calculating the positive and negative tweet sentiments out of the tweets posted by the user. When the agent was successful in analyzing the tweet sentiments, it was modified to calculate the total number of positive or negative sentiments in the whole tweet set. In this way, agent could infer about the in general tweet sentiments from the tweet dataset.

Next, the topic assigner agent's knowledge was further filtered using the date range or hashtag. The processing of tweets is similar to before. Just when filters of date and hashtags are applied, the topic assigner agent will narrow down the results in the way shown in fig 18 and fig 19.