Background Knowledge Aware Semantic Coding Model Selection

Fangzhou Zhao, Yao Sun*, Runze Cheng, Muhammad Ali Imran
James Watt School of Engineering, University of Glasgow, Glasgow, UK
f.zhao.1@research.gla.ac.uk, {Yao.Sun, Muhammad.Imran}@glasgow.ac.uk

***Abstract*—Semantic communication is deemed to break Shannon channel capacity by transmitting extracted semantics rather than all binary bits. One critical challenge in semantic communication system is how to select a matching semantic coding model (SCM) in light of complicated source information, diversified user background knowledge (BK) and dynamic wireless channel. In this paper, we mathematically model the relationship among different BKs by using graph theory, and introduce a metric to evaluate SCMs performance as per BK relationships. Then, we propose a Background knowledge Aware SCM SElection (BASE) scheme, where a deep learning algorithm is exploited to accurately predict SCM performance in context of the modeled BK, guiding the SCM selection. Numerical simulation results show that the BASE has superiorities in information recovery accuracy along with the probability of selecting the optimal SCM when compared with other benchmarks.**

***Index Terms*—Semantic communication, Semantic coding model selection, Background knowledge, Deep learning**

I. INTRODUCTION

There is a potential revolution coming to wireless communication system with the goal of transmitting semantics extracted by machine learning instead of all binary bits for each information [1]. This novel paradigm is called *semantic communication* [2], [3], where semantic encoders extract semantics from source information for transmission, and semantic decoders recover the meaning of source information based on the received semantics. In this way, transmission reliability can be improved, especially under an unsatisfactory wireless channel condition with high bit-error rate. More importantly, wireless communication resource consumption can be greatly reduced with much less binary bits transmitted in semantic communication system [4]–[6].

In particular, there are a certain number of semantic coding models (SCMs) in semantic communication system, and the selection of an appropriate SCM for source information can greatly affect the information recovery accuracy [7], [8]. For example, if an SCM trained by quantum physics literature datasets is used to encode and decode for the communication scenario of classical literature transmission, the information recovery accuracy should be extremely unsatisfactory due to the high mismatch of background knowledge (BK) between the SCM and the source information. However, the biggest challenge in SCM selection is that the actual performance of an SCM cannot be directly measured before transmission starts. It is even more difficult considering that multiple similar SCMs could exist for the same service type.

Inspired by the above, we hypothesize that BKs of both SCM and source information should be taken into account for guiding SCM selection. Basically, the BK of source information represents a specific user behavior including preferred contents, frequent communication time and regions. The BK of an SCM is constructed during its training process, where it is updated by all the BKs of training samples collected from the practical transmission. Intuitively, if the two BKs are similar, which means that the training samples used for the SCM should be similar to the source information, the SCM could achieve a better performance on extracting and recovering semantics for this source information. However, the difficulty in using BK to guide SCM selection is how to precisely examine the similarity between the two BKs. Therefore, modelling the relationship between BKs should be the key in SCM selection.

Knowledge modelling has been extensively investigated in other areas such as content searching systems [9], recommendation systems [10] and database indexing [11]. However, there is no pioneer work addressing how to measure the relationship between BKs in semantic communication. Unfortunately, the knowledge modeling in other fields cannot be directly applied to semantic communication due to that the information of each BK in semantic communication is massive, dynamic and diverse. Even more difficult is the various information in BK has different and dynamic importance for the description of BK.

In this paper, we investigate SCM selection, a fundamental yet challenging issue for semantic communication, and propose a Background knowledge Aware SCM SElection (BASE) scheme based on deep learning. The main contributions of this paper are summarized as follows.

- *BK modeling for semantic communication system:* We divide individual BK into multiple knowledge tiles and model the relations between knowledge tiles as a weighted undirected graph. Thus the relation between BKs can be calculated based on the graph.
- *Defining new performance metric for SCM:* A new metric BK similarity is defined to measure the similarity between the BK of SCM and that of the source information.
- *Designing BK aware SCM selection:* A deep learning algorithm is used in the BASE to accurately predict the SCM performance ranking by inputting the above BK

*Yao Sun is corresponding author.

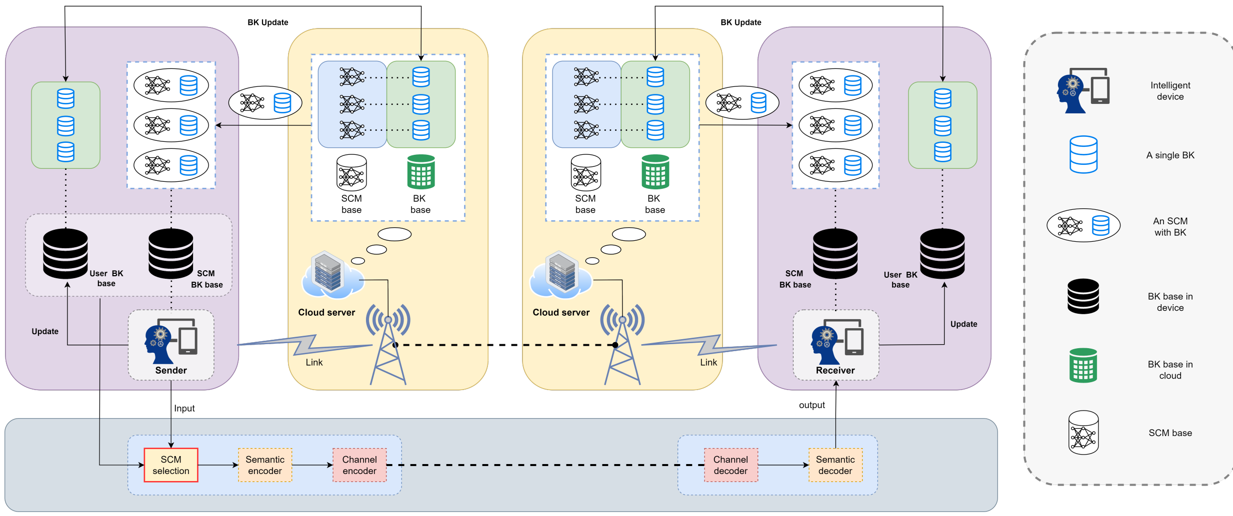


Fig. 1. The framework of semantic communication

similarities. Based on this ranking, the optimal SCM can be selected before transmission starts.

- *Conducting simulation verification:* The numerical simulation results show that the proposed BASE can effectively improve the probability of selecting the optimal SCM and achieve a better semantic accuracy compared with other benchmarks.

The remainder of this paper is organized as follows. In Section II, we introduce the system model and give the problem description. In Section III, we propose the solution of the SCM selection based on deep learning. In Section IV, we present our numerical simulation results and conclude this paper in Section V.

II. System model and Problem formulation

We consider an intelligent end-to-end semantic communication system consisting of base stations (BSs) along with cloud servers and devicesas shown in Fig. 1. Specifically, there is one global SCM base on each BS containing all the required SCMs for users in this coverage. Each SCM is associated with a specific BK (called SCM BK) trained and updated by the data generated from all users. At the device end, it downloads and stores the preferred SCMs and their associated SCM BKs from the BS, forming the local SCM base and local SCM BK base, as shown in Fig. 1. Besides, each device stores all the BKs of source information sent by the user in a history period (called source information BK), and they are maintained by the knowledge base on the device side. Please note that all the SCMs and the BKs should be periodically updated with the help of cloud servers, and the update policy is out of the scope of this work. Therefore, we investigate the SCM selection under a static scenario with no updates.

Let us briefly streamline the process of end-to-end semantic communication. As shown in Fig. 1, the sender first chooses an appropriate SCM based on BK for semantic encoding. Once the SCM is selected, the entire encoder (including semantic and channel encoders) is determined. Then, the source information goes through the encoder to extract semantic information. Finally, the semantics are packed into binary bits and transmitted through wireless channels. At the receiver end, it first selects the same SCM used by the sender, and then the received bits are reshaped to semantics through the channel decoder, and finally the semantics are recovered to the requested information with the help of BK.

A. Model of SCM BK and Source Information BK

We assume both SCM BK and source information BK are in the same structure but generated by different data. As discussed above, source information BK representing a specific user behavior is maintained by this user's transmission status while SCM BK is maintained by the training data from all users. Let us mathematically model source information BKs and SCM BKs. As shown in Fig.2, denote $\mathscr{B}^{S_i} = \left[\left(U_1^{S_i}, U_2^{S_i}, .., U_L^{S_i}\right); \left(V_1^{S_i}, V_2^{S_i}, .., V_l^{S_i}\right)\right]$ and $\mathscr{B}^{M_j} = \left[\left(U_1^{M_j}, U_2^{M_j}, .., U_L^{M_j}\right); \left(V_1^{M_j}, V_2^{M_j}, .., V_l^{M_j}\right)\right]$ as the BK of source information S_i and the SCM M_j, respectively, where $U_k^{S_i}$ (or $U_k^{M_j}$) is the kth long-term knowledge tile of source information S_i (or SCM M_j), and $V_h^{S_i}$ (or $V_h^{M_j}$) is the hth short-term knowledge tile of source information S_i (or SCM M_j). Especially, a long-term knowledge tile describes a specific user behavior, such as a preferred topic; while a short-term knowledge tile describes a part of the real-time status of the user, such as location. Therefore, the relations between BKs can be measured by the knowledge tile relations.

To describe the relationships between knowledge tiles, we exploit graph theory as the tool. As shown in Fig. 3, we consider each knowledge tile as a vertex in a graph and the relationship between tiles $U_k^{S_i}$ and $U_k^{M_j}$ as an edge with a value $\lambda_{U_k}^{ij}$ showing how close between the two tiles is. Some knowledge tiles are not connected if there is no direct relation between them. The $\lambda_{U_k}^{ij}$ between two unconnected tiles is the

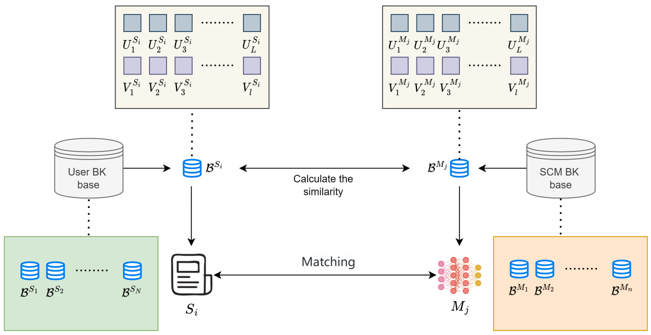


Fig. 2. The structure of SCM BK (and source information BK)

sum of edge values for the shortest path between the two vertexes.

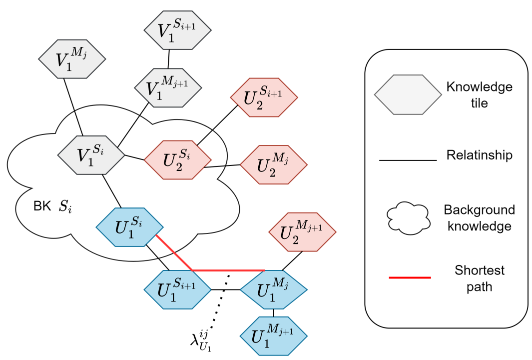


Fig. 3. Knowledge tiles relationships graph

Importantly, we calculate the BK similarity $r_{S_i}^{M_j}$ between the source information BK S_i and SCM BK M_j based on the relation of tiles,

$$r_{S_i}^{M_j} = \sum_{k=1}^{L} \frac{\omega_k^{ij} \lambda_{U_k}^{ij}}{L} + \sum_{k'=1}^{l} \frac{\sigma_{k'}^{ij} \lambda_{V_{k'}}^{ij}}{l}, \tag{1}$$

where ω_k^{ij} represents the weight of the relationship between long-term tiles $U_k^{S_i}$ and $U_k^{M_j}$, and $\sigma_{k'}^{ij}$ represents the weight of the relationship between short-time tiles $V_{k'}^{S_i}$ and $V_{k'}^{M_j}$.

B. Metric of SCM Performance

We use semantic accuracy to measure the performance of SCM and define semantic accuracy as $E_{S_i}^{M_j}$ when using SCM M_j to code source information S_i. Specially, the calculation of semantic accuracy depends on the service type, e.g., using bilingual evaluation understudy for text messages and structural similarity index measure for image message.

We sort the semantic accuracy of all SCMs for a given source information S_i, which can be expressed as

$$O_{S_i} = Sort\left(E_{S_i}^{M_1}, E_{S_i}^{M_2}, ..., E_{S_i}^{M_n}\right) = \left(o_{S_i}^1, o_{S_i}^2, ..., o_{S_i}^n\right), \tag{2}$$

where $o_{S_i}^a$ represents the SCM with the ath highest semantic accuracy. We sort the BK similarity of all SCMs for a given source information S_i as

$$\hat{O}_{S_i} = Sort\left(r_{S_i}^{M_1}, r_{S_i}^{M_2}, ..., r_{S_i}^{M_n}\right) = \left(\hat{o}_{S_i}^1, \hat{o}_{S_i}^2, ..., \hat{o}_{S_i}^n\right), \tag{3}$$

where $\hat{o}_{S_i}^a$ represents the SCM with the ath highest BK similarity.

Based on the above two rankings, we define knowledge ambiguity $G_{S_i}\left(\omega_k^{ij}, \sigma_{k'}^{ij}\right)$ as

$$G_{S_i}\left(\omega_k^{ij}, \sigma_{k'}^{ij}\right) = \frac{\sum_{a=1}^{n} D\left(o_{S_i}^a, \hat{o}_{S_i}^a\right)}{n}, \tag{4}$$

where

$$D\left(o_{S_i}^a, \hat{o}_{S_i}^a\right) = \begin{cases} 1, & o_{S_i}^a = \hat{o}_{S_i}^a \\ 0, & o_{S_i}^a \neq \hat{o}_{S_i}^a \end{cases}, \tag{5}$$

indicating whether the ath ranking is the same SCM for semantic accuracy and BK similarity. The higher the value of G_{S_i}, the closer the BK similarity between the actual SCM performance.

C. Problem Description

As expounded above, to maximize G_{S_i}, a deep neural network (DNN) framework is used to accurately predict SCM performance. By inputting the knowledge tile relationships from history transmission records into the neural network and using the performance ranking of the SCMs as supervision, the performance ranking of SCM based on BK similarity should converge to the semantic accuracy ranking. The loss function of this DNN uses cross-entropy to measure the difference between semantic accuracy ranking and BK similarity ranking. For give $\mathscr{B}^{S_i}$ and $\mathscr{B}^{M_j}$ let us denote $P^{ij} = \left[\left(\lambda_{U_1}^{ij}, \lambda_{U_2}^{ij}, .., \lambda_{U_L}^{ij}\right); \left(\lambda_{V_1}^{ij}, \lambda_{V_2}^{ij}, .., \lambda_{V_l}^{ij}\right)\right]$ as the set of all path values between each pair of knowledge tiles and the set of all P^{ij} is denoted as $\mathscr{P}$. Thus, the loss function can be denoted as

$$L\left(\mathscr{P}, O_{S_i}, \omega, \sigma\right) = -\sum_{j=1} \sum_{c=1}^{l} p_c\left(\mathscr{P}^{ij}\right) \log\left(q_c\left(\mathscr{P}^{ij}\right)\right), \tag{6}$$

where $p_c\left(\mathscr{P}^{ij}\right)$ is the real probability that SCM M_j is at the cth position in semantic accuracy ranking, and $q_c\left(\mathscr{P}^{ij}\right)$ is the predicted probability that the SCM M_j is at cth position in semantic accuracy ranking. By reducing the loss value $q_c\left(\mathscr{P}^{ij}\right)$ can be maximized, thus approaching $p_c\left(\mathscr{P}^{ij}\right)$.

III. Proposed BASE scheme

In this section, we propose the BASE to select appropriate SCM. Specifically, we take the relationship between knowledge tiles as the input and the output is a probability distribution of selecting the optimal SCM. Hence, the SCM with the highest probability in the output should be selected to perform encoding/decoding for a given source information.

The proposed BASE is shown in the Fig. 4, where the input data consists of $\{\lambda_{U_1}^{ij}, \lambda_{U_2}^{ij}, ..., \lambda_{U_L}^{ij}\}$ and $\{\lambda_{V_1}^{ij}, \lambda_{V_2}^{ij}, ..., \lambda_{V_l}^{ij}\}$. In order to feed the data into the neural network, embedding is required first to convert the input data into binary-format. Hence,

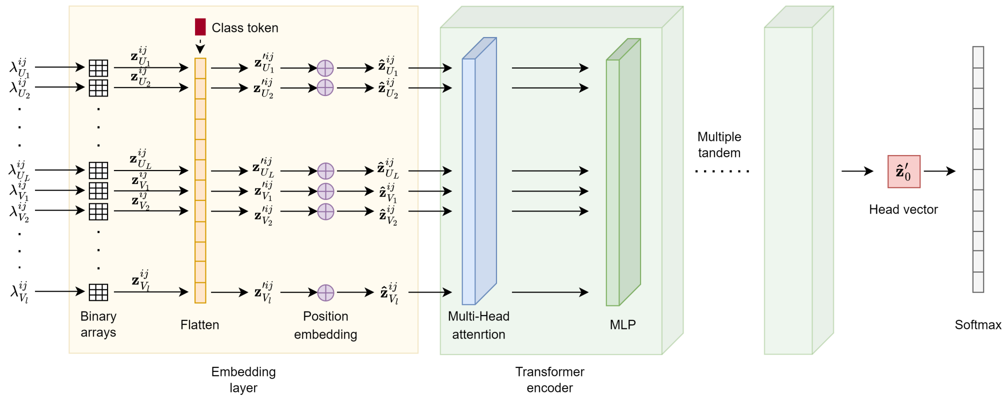


Fig. 4. The diagram of ViT based model selector

we convert $\{\lambda^{ij}_{U_1}, \lambda^{ij}_{U_2}, ..., \lambda^{ij}_{U_L}\}$ and $\{\lambda^{ij}_{V_1}, \lambda^{ij}_{V_2}, ..., \lambda^{ij}_{V_l}\}$ into the binary arrays $\left[\mathbf{z}^{ij}_{U_1}, \mathbf{z}^{ij}_{U_2}, ..., \mathbf{z}^{ij}_{U_L}\right]$ and $\left[\mathbf{z}^{ij}_{V_1}, \mathbf{z}^{ij}_{V_2}, ..., \mathbf{z}^{ij}_{V_l}\right]$ which contains fixed-length binary numbers.

The proposed BASE requires the use of a standard transformer encoder with a one-dimensional sequence of marker embeddings as input. Therefore, we input the binary arrays $\left[\mathbf{z}^{ij}_{U_1}, \mathbf{z}^{ij}_{U_2}, ..., \mathbf{z}^{ij}_{U_L}\right]$ and $\left[\mathbf{z}^{ij}_{V_1}, \mathbf{z}^{ij}_{V_2}, ..., \mathbf{z}^{ij}_{V_l}\right]$ into a fully-connected layer to convert them into the vectors $\left[\mathbf{z}'^{ij}_{U_1}, \mathbf{z}'^{ij}_{U_2}, ..., \mathbf{z}'^{ij}_{U_L}\right]$ and $\left[\mathbf{z}'^{ij}_{V_1}, \mathbf{z}'^{ij}_{V_2}, ..., \mathbf{z}'^{ij}_{V_l}\right]$, thus satisfying the input requirement of one-dimensional sequence in transformer. Since multiple embedded vectors should be input to the transformer at the same time, it is necessary to embed the front-to-back relationship among these vectors, and this process is called positional embedding (PE). The PE method used in the BASE is similar to that of Vision Transformer [12]. The output of the PE is denoted as $\left[\hat{\mathbf{z}}^{ij}_{U_1}, \hat{\mathbf{z}}^{ij}_{U_2}, ..., \hat{\mathbf{z}}^{ij}_{U_L}\right]$ and $\left[\hat{\mathbf{z}}^{ij}_{V_1}, \hat{\mathbf{z}}^{ij}_{V_2}, ..., \hat{\mathbf{z}}^{ij}_{V_l}\right]$, and the embedding layer ends here.

Once the embedding layer is passed, embedded vectors are loaded into the transformer encoder. In particular, the transformer encoder consists of a superposition of identical structures containing a multi-headed self-attention (MSA) and a feed-forward neural network (FFN). Self-attention (SA) is the key structure of the transformer which filters out key information and makes DNN focus on that important information. In transformer, there is an MSA enabling the DNN to view the previous input. In MSA, the input features are multiplied with different transformation matrices to obtain the three vector $\mathbf{q}_z, \mathbf{k}_z, \mathbf{v}_z$, and MSA extracts the input information based on the elements in these three vectors groups [13]. When the input vector is $\hat{\mathbf{z}}^{ij}$, the self-attention can be expressed as

$$A\left(\hat{\mathbf{z}}^{ij}\right) = \mathrm{softmax}\left(\frac{\mathbf{q}_z {\mathbf{k}_z}^T}{\sqrt{|\mathbf{k}_z|}}\right)\mathbf{v}_z, \tag{7}$$

where ${\mathbf{k}_z}^T$ is the transpose of $\mathbf{k}_z$ and $|\mathbf{k}_z|$ is the dimension. Thus, the MSA can be expressed as

$$M\left(\hat{\mathbf{z}}^{ij}\right) = concat\left(A_1\left(\hat{\mathbf{z}}^{ij}\right), A_2\left(\hat{\mathbf{z}}^{ij}\right), .., A_n\left(\hat{\mathbf{z}}^{ij}\right)\right)W, \tag{8}$$

where W is the parameter matrix for projection, and function $concat$ is used to concatenate multiple SAs with the same calculation rule defined in [13].

For the output of the transformer encoder, only the first output vector of the encoder (the head vector denoted as $\hat{\mathbf{z}}'_0$) needs to be exploited. This is because the head vector of the transformer contains the required information of all other output vectors [14].

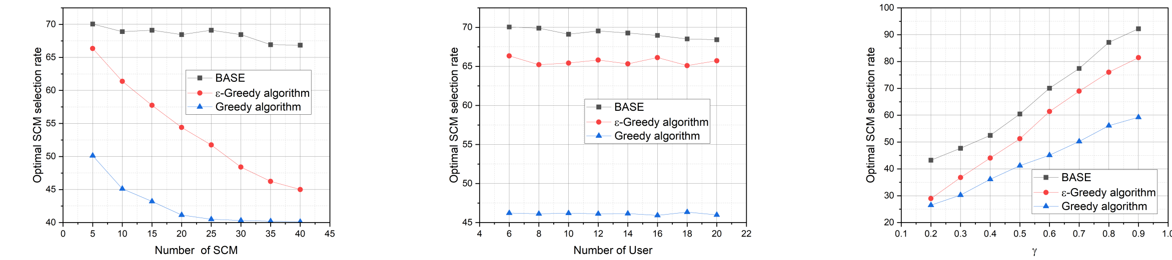
The Softmax function is then used to process the head vector $\hat{\mathbf{z}}'_0$ to obtain the probability distribution of the performance ranking of the SCM M_j, which is denoted as $Pr(M_j)$. The SCM with the highest probability of ranking first is the optimal model that should be selected for encoding and decoding this source information S_i.

IV. Results and Discussions

In this section, we evaluate the performance of the proposed SCM selection scheme BASE by comparing with other benchmarks. Two benchmarks epsilon-greedy algorithm [15] and greedy algorithm are used. The epsilon-greedy algorithm selects the highest ranked SCM in terms of BK similarity with a probability of $1 - \epsilon$, and randomly selects the other top 3 ranked models with a probability of ϵ, which is set to 0.2 in our simulations. Another benchmark greedy algorithm always selects the SCM with the highest BK similarity.

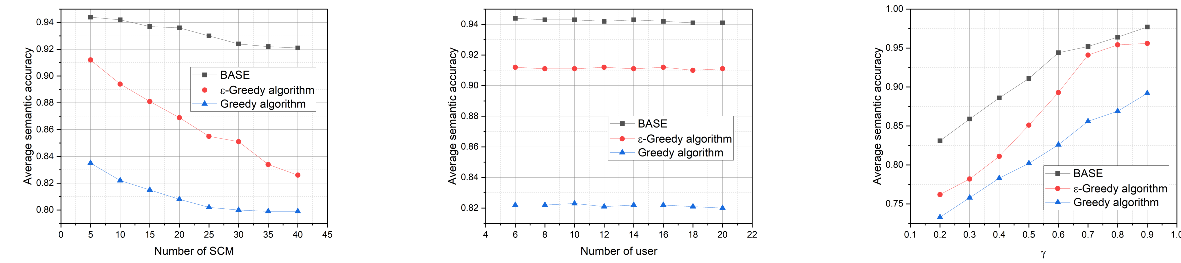
A. Simulation Settings

In the simulations, we use numbers to represent knowledge tiles, and each training sample is consisted of three long-term knowledge tiles and two short-term knowledge tiles. Labels in samples contain the semantic accuracy and semantic accuracy rankings of SCMs. In particular, some BK samples used for specific SCMs do not have label. In order to consider a more realistic scenario, two of the knowledge tiles representing long-term knowledge are set to randomly initialized fixed values, and the remaining one is set as a dynamic variable. In particular, a certain probability that the short-term knowledge tiles are set as discrete random variables and the rest probability are set as continuous random variables, this probability is denoted as γ.



(a) Optimal SCM selection rate versus number of SCM (b) Optimal SCM selection rate versus number of user (c) Optimal SCM selection rate versus γ

Fig. 6. Comparisons of optimal SCM selection rate for the three SCM selection schemes



(a) Average semantic accuracy versus number of SCM (b) Average semantic accuracy versus number of user (c) Average semantic accuracy versus γ

Fig. 7. Comparisons of average semantic accuracy for the three SCM selection schemes

Parameters of the DNN model used in the simulation are shown in Table 1.

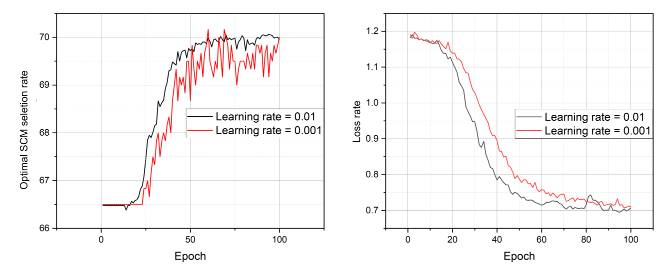
TABLE I
THE SETTINGS OF THE MODEL SELECTION NETWORK

Layer name	Unit	Activation
Vit Encoder	128 (8 heads)	Linear
Denses	128 (6 depth)	Relu
Mlp	128	Linear
Prediction layer	Size of Model Base	Softmax

In the simulation, the transformer has 6 blocks, with 8 heads and 128 units, the Dense layer has 128 units and 6 depth respectively, and the Mlp layer has 128 units. The default SCM number is set to 10, the default users number is set to 6, and the default value of γ is set to 0.6. The simulations are carried out by the computer equipped with Intel Core i7-11700F CPU@2.50 GHz and NVIDIA GeForce GTX3060.

B. Simulation Results

We first examine the convergence of DNN used in the BASE, as shown in Fig. 5. From Fig. 5(a), both the optimal SCM selection rate with the two learning rates (0.01 and 0.001) converges to 70% after about 50 epochs, and the curve of the learning rate 0.01 converges with less oscillation. In Fig. 5(b), the two curves of the loss value at the two learning rates converge after about 50 epochs, and the curve of the learning rate 0.01 converges faster. Thus, the learning rate in the remaining simulations is set to 0.01.



(a) The optimal SCM selection rate versus epoch (b) The loss value versus epoch

Fig. 5. The optimal SCM selection rate and loss value versus epoch in different learning rate

Next, we compare the optimal SCM selection rate of the three schemes under different scenarios shown in Fig. 6. In Fig. 6(a), we find that the rate of the BASE remains stable, while the rate of the other two benchmarks decreases rapidly. This is because the deep learning used in the BASE predicts the SCM performance accurately, which improves the optimal SCM selection rate. In Fig. 6(b), the rate of the three scenarios varies slightly with the number of users. This is because changing the number of users does not affect the overall BK structure so that the SCM performance. Moreover, in Fig. 6(c), all the three schemes increase with γ, while the greedy algorithm increases more slowly. This is because the number of discrete random variables in BK increases with γ, which reduces the feasible region for BK and makes it less difficult to select the optimal SCM. In particular, the rational that the

BASE outperforms greedy algorithm is DNN used explores more possibilities in the BK feasible region.

At last, the average semantic accuracy comparisons of the three schemes under different scenarios are presented in Fig. 7. The BASE achieves the highest average semantic accuracy among the three schemes under all the three scenarios. This indicates that the BK modeling derived in this work is reasonable and the BASE scheme can accurately predict the performance of the SCM. In particular, the BASE maintains a high average semantic accuracy under all the three scenarios, which implies that the BASE is able to select a sub-optimal SCMs even if it cannot select the optimal one.

V. Conclusion

In this paper, we investigate the problem of SCM selection in semantic communications. Specifically, BKs are modelled using a graph-theoretic, the similarity between SCMs and source information are measured by relationships among BKs, and the BASE scheme is proposed which applying deep learning to more accurately predict SCM performance to guide SCM selection. The simulation results demonstrate the BASE outperforms other benchmarks and the reliability of semantic communication can be improved by SCM selection. This work can be seen as a pioneer of optimizing SCM selection in semantic communication, which reveals another angle to enhance the information recovery accuracy. Moreover, this work provides a foundation for subsequent research on SCM selection under more complex scenarios.

References

[1] E. C. Strinati and S. Barbarossa, "6g networks: Beyond shannon towards semantic and goal-oriented communications," *Computer Networks*, vol. 190, p. 107930, 2021.

[2] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021.

[3] Q. Lan, D. Wen, Z. Zhang, Q. Zeng, X. Chen, P. Popovski, and K. Huang, "What is semantic communication? a view on conveying meaning in the era of machine intelligence," *Journal of Communications and Information Networks*, vol. 6, no. 4, pp. 336–371, 2021.

[4] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, "Edge artificial intelligence for 6g: Vision, enabling technologies, and applications," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 5–36, 2021.

[5] L. Xia, Y. Sun, X. Li, G. Feng, and M. A. Imran, "Wireless resource management in intelligent semantic communication networks," *arXiv preprint arXiv:2202.07632*, 2022.

[6] P. Zhang, W. Xu, H. Gao, K. Niu, X. Xu, X. Qin, C. Yuan, Z. Qin, H. Zhao, J. Wei *et al.*, "Toward wisdom-evolutionary and primitive-concise 6g: A new paradigm of semantic communication networks," *Engineering*, vol. 8, pp. 60–73, 2022.

[7] G. Shi, Y. Xiao, Y. Li, and X. Xie, "From semantic communication to semantic-aware networking: Model, architecture, and open problems," *IEEE Communications Magazine*, vol. 59, no. 8, pp. 44–50, 2021.

[8] Y. Yang, C. Guo, F. Liu, C. Liu, L. Sun, Q. Sun, and J. Chen, "Semantic communications with AI tasks," *CoRR*, vol. abs/2109.14170, 2021. [Online]. Available: https://arxiv.org/abs/2109.14170

[9] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A large ontology from wikipedia and wordnet," *Journal of Web Semantics*, vol. 6, no. 3, pp. 203–217, 2008, world Wide Web Conference 2007Semantic Web Track. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570826808000437

[10] A. Neelakantan, B. Roth, and A. McCallum, "Compositional vector space models for knowledge base completion," *CoRR*, vol. abs/1504.06662, 2015. [Online]. Available: http://arxiv.org/abs/1504.06662

[11] W. Zhang, B. Paudel, L. Wang, J. Chen, H. Zhu, W. Zhang, A. Bernstein, and H. Chen, "Iteratively learning embeddings and rules for knowledge graph reasoning," in *The World Wide Web Conference*, ser. WWW '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2366–2377. [Online]. Available: https://doi.org/10.1145/3308558.3313612

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *CoRR*, vol. abs/2010.11929, 2020. [Online]. Available: https://arxiv.org/abs/2010.11929

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[14] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805

[15] A. Slivkins, "Introduction to multi-armed bandits," *CoRR*, vol. abs/1904.07272, 2019. [Online]. Available: http://arxiv.org/abs/1904.07272